

E5653

PATENT ABSTRACTS OF JAPAN



(11)Publication number : 10-307808
 (43)Date of publication of application : 17.11.1998

(51)Int.Cl.

G06F 17/00

(21)Application number : 09-116794
 (22)Date of filing : 07.05.1997

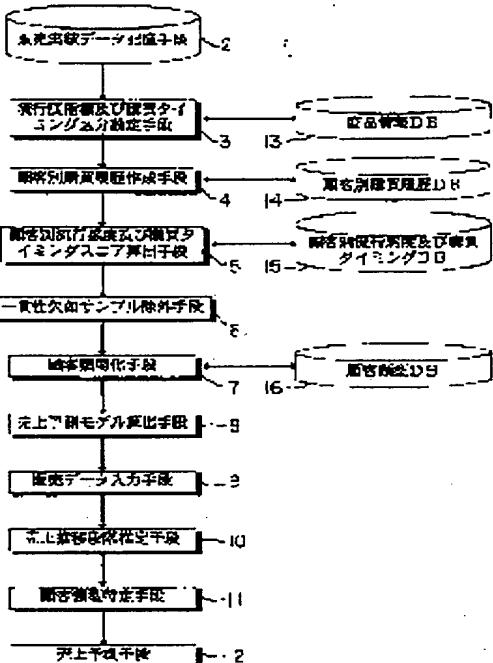
(71)Applicant : NRI & NCC CO LTD
 (72)Inventor : TORIYAMA MASAHIRO

(54) SALES PREDICTION DEVICE CONSIDERING POPULARITY PREDICTION OF PRODUCT AND PREDICTION METHOD THEREFOR

(57)Abstract:

PROBLEM TO BE SOLVED: To quickly reflect the popularity of products on a production sales schedule by recognizing the tendency of the type, constitution and sales transition of persons who purchase the product in the initial stage of the sale of the product and predicting the sales of the product.

SOLUTION: A sales data input means 9 sends the sales data of the product to this sales prediction device 1 preferably in real time. Then, by a sales transition stage estimation means 10, the sales transition stage of the product is estimated. Then, a customer type specifying means 11 inputs customer data inputted by the sales data input means 9, retrieves a customer type data base 16 for respective customers and specifies the customer type from the pertinent customer data. A sales prediction means 12 inputs the sales transition stage estimated by the sales transition stage estimation means 10, the customer type specified by the customer type specifying means 11 and a sales prediction model calculated by a sales prediction model calculation means 8 and predicts the sales of a prediction object product or the degree of the popularity by using the data and a numerical formula, etc.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

THIS PAGE BLANK (USPTO)

モチ653

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平10-307808

(43)公開日 平成10年(1998)11月17日

(51)Int.Cl.⁶
G 0 6 F 17/00

識別記号

F I
G 0 6 F 15/20

F

審査請求 未請求 請求項の数6 O.L (全12頁)

(21)出願番号 特願平9-116794

(22)出願日 平成9年(1997)5月7日

(71)出願人 000155469

株式会社野村総合研究所
東京都中央区日本橋1丁目10番1号(72)発明者 島山正博
神奈川県横浜市保土ヶ谷区神戸町134番地
株式会社野村総合研究所内

(74)代理人 弁理士 佐藤一雄 (外3名)

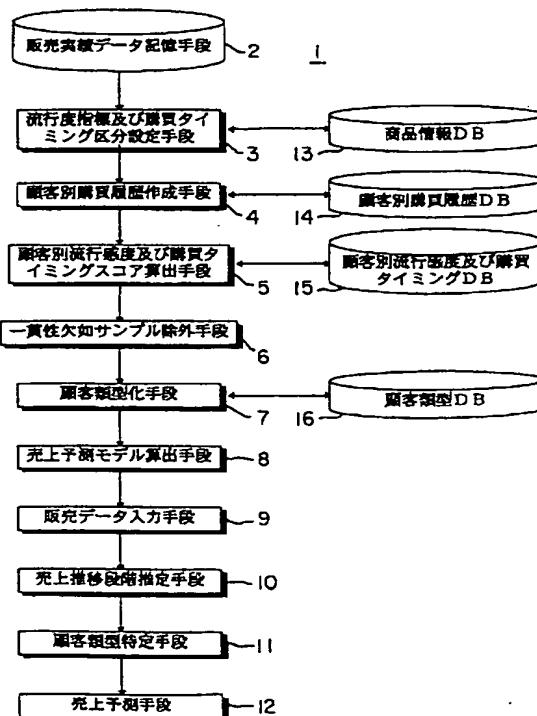
スコア化
売上予測

(54)【発明の名称】商品の流行予測を加味した売上予測装置及びその予測方法

(57)【要約】

【課題】商品の販売の初期段階でその商品を購買した者の類型、構成、売上推移の動向から商品の流行予測を加味した売上予測装置及びその予測方法を提供する。

【解決手段】販売実績データ記憶手段2と、流行度指標及び購買タイミング区分設定手段3と、顧客別購買履歴作成手段4と、顧客別流行感度及び購買タイミングスコア算出手段5と、一貫性欠如サンプル除外手段6と、顧客類型化手段7と、顧客の類型から商品の売上予測モデルを算出する売上予測モデル算出手段8と、販売データ入力手段9と、売上推移段階推定手段10と、顧客類型特定手段11と、売上推移段階推定手段10が推定した売上推移段階と、顧客類型特定手段11が特定した顧客類型と、売上予測モデル算出手段8が算出した売上予測モデルとから商品の売上予測を出力する売上予測手段12と、を備えた。



【特許請求の範囲】

【請求項1】商品の品目、商品を購買した顧客、商品が販売された日時を含む商品の販売実績データを格納する販売実績データ記憶手段と、

前記販売実績データ記憶手段から商品の販売実績データを検索して入力し、商品ごとに流行度指標と購買タイミング区分とを設定し、商品情報データベースに出力する流行度指標及び購買タイミング区分設定手段と、

前記販売実績データ記憶手段から顧客の購買実績データを検索して入力し、顧客別に購買した商品の品目、購買時点を整理し、顧客別購買履歴データベースに出力する顧客別購買履歴作成手段と、

前記顧客別購買履歴データベースから顧客別の購買履歴データを入力し、各顧客が購買した商品の全品目について前記商品情報データベースを参照し、各購買商品の流行度指標と購買時点が属する購買タイミング区分に応じてそれぞれ流行度スコアと購買タイミングスコアを付与し、前記流行度スコアと購買タイミングスコアの平均値またはモード値、標準偏差値または分散値を算出し、顧客別流行度及び購買タイミングデータベースに出力する顧客別流行度及び購買タイミングスコア算出手段と、

前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの標準偏差値または分散値を入力し、前記標準偏差値または分散値により、流行度及び購買タイミングに一貫性を欠く顧客のデータを前記顧客別流行度及び購買タイミングデータベースから除外する一貫性欠如サンプル除外手段と、

前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの平均値またはモード値を入力し、流行度スコアの平均値またはモード値、購買タイミングスコアの平均値またはモード値の組合せにより、流行に対する感度と商品の購買タイミングを因子とする複数の顧客類型に顧客を分類し、顧客類型データベースに出力する顧客類型化手段と、

過去の商品の販売実績データを入力し、商品の売上推移の段階と、商品の各売上推移段階中に商品を購買した顧客の類型の構成比とを説明変数とし、その商品の売上げまたは流行の度合いを被説明変数として、商品の売上予測モデルを算出する売上予測モデル算出手段と、

予測対象商品について、その商品を購買した顧客と、商品が販売された日時のデータを入力する販売データ入力手段と、

前記販売データ入力手段が入力した商品の販売データを入力し、売上推移の傾斜から商品の売上推移段階を推定する売上推移段階推定手段と、

前記販売データ入力手段が入力した顧客データを入力し、これらの顧客データの各顧客について前記顧客類型

データベースを参照することにより、各顧客の顧客類型を特定する顧客類型特定手段と、

前記売上推移段階推定手段が推定した売上推移段階と、前記顧客類型特定手段が特定した顧客類型と、前記売上予測モデル算出手段が算出した売上予測モデルとを入力し、前記予測対象商品の売上げまたは流行の度合いを予測して出力する売上予測手段と、を備えたことを特徴とする商品の流行予測を加味した売上予測装置。

【請求項2】前記販売実績データ記憶手段は、商品のカテゴリーのデータを格納しており、

前記顧客別購買履歴作成手段は、商品のカテゴリーごとに、各顧客が購買した商品の品目、購買時点を整理し、顧客別購買履歴データベースに出力し、

前記顧客別流行度及び購買タイミングスコア算出手段は、各商品のカテゴリーごとに、前記流行度スコアと購買タイミングスコアの平均値またはモード値、標準偏差値または分散値を算出し、顧客別流行度及び購買タイミングデータベースに出力し、

前記一貫性欠如サンプル除外手段は、各商品のカテゴリーごとに、流行度及び購買タイミングに一貫性を欠く顧客のデータを前記顧客別流行度及び購買タイミングデータベースから除外し、前記顧客類型化手段は、各商品のカテゴリーごとに、流行に対する感度と商品の購買タイミングを因子とする複数の顧客類型に顧客を分類し、顧客類型データベースに出力し、

前記売上予測モデル算出手段は、各商品のカテゴリーごとに、商品の売上予測モデルを算出し、

前記売上予測手段は、商品カテゴリーに対応する売上予測モデルを用いて前記予測対象商品の売上げまたは流行の度合いを予測する、ように構成されていることを特徴とする請求項1に記載の商品の流行予測を加味した売上予測装置。

【請求項3】前記販売データ入力手段は、商品を購買した顧客と、商品が販売された日時のデータをリアルタイムで入力し、

前記売上予測手段は、前記販売データ入力手段がリアルタイムで入力したデータを用いて商品の売上げまたは流行の度合いを予測する、ように構成されていることを特徴とする請求項1または2に記載の商品の流行予測を加味した売上予測装置。

【請求項4】商品の品目、商品を購買した顧客、商品が販売された日時を含む商品の販売実績データを格納した販売実績データ記憶手段から商品の販売実績データを検索して入力し、商品ごとに流行度指標と購買タイミング区分とを設定し、商品情報データベースに出力し、

前記販売実績データ記憶手段から顧客の購買実績データを検索して入力し、顧客別に購買した商品の品目、購買時点を整理し、顧客別購買履歴データベースに出力し、前記顧客別購買履歴データベースから顧客別の購買履歴データを入力し、各顧客が購買した商品の全品目につい

て前記商品情報データベースを参照し、各購買商品の流行度指標と購買時点が属する購買タイミング区分に応じてそれぞれ流行度スコアと購買タイミングスコアを付与し、前記流行度スコアと購買タイミングスコアの平均値またはモード値、標準偏差値または分散値を算出し、顧客別流行度及び購買タイミングデータベースに出力し、

前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの標準偏差値または分散値を入力し、前記標準偏差値または分散値により、流行度及び購買タイミングに一貫性を欠く顧客のデータを前記顧客別流行度及び購買タイミングデータベースから除外し、

前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの平均値またはモード値を入力し、流行度スコアの平均値またはモード値、購買タイミングスコアの平均値またはモード値の組合せにより、流行に対する感度と商品の購買タイミングを因子とする複数の顧客類型に顧客を分類し、顧客類型データベースに出力し、

過去の商品の販売実績データを入力し、商品の売上推移の段階と、商品の各売上推移段階中に商品を購買した顧客の類型の構成比とを説明変数とし、その商品の売上げまたは流行の度合いを被説明変数として、商品の売上予測モデルを算出し、

予測対象商品の販売データを入力し、売上推移の傾斜から商品の売上推移段階を推定し、

予測対象商品の顧客データを入力し、これらの顧客データの各顧客について前記顧客類型データベースを参照することにより、各顧客の顧客類型を特定し、

前記推定した売上推移段階と、前記特定した顧客類型と、前記算出した売上予測モデルとを入力し、前記予測対象商品の売上げまたは流行の度合いを予測することを特徴とする商品の流行予測を加味した売上予測方法。

【請求項5】商品の品目、カテゴリー、商品を購買した顧客、商品が販売された日時を含む商品の販売実績データを格納した販売実績データ記憶手段から商品の販売実績データを検索して入力し、商品ごとに流行度指標と購買タイミング区分とを設定し、商品情報データベースに出力し、

前記販売実績データ記憶手段から顧客の購買実績データを検索して入力し、商品のカテゴリーごとに、顧客別に購買した商品の品目、購買時点を整理し、顧客別購買履歴データベースに出力し、

前記顧客別購買履歴データベースから顧客別の購買履歴データを入力し、商品のカテゴリーごとに、各顧客が購買した商品の全品目について前記商品情報データベースを参照し、各購買商品の流行度指標と購買時点が属する購買タイミング区分に応じてそれぞれ流行度スコアと購買タイミングスコアを付与し、前記流行度スコアと

購買タイミングスコアの平均値またはモード値、標準偏差値または分散値を算出し、顧客別流行度及び購買タイミングデータベースに出力し、

前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの標準偏差値または分散値を入力し、商品のカテゴリーごとに、前記標準偏差値または分散値により、流行度及び購買タイミングに一貫性を欠く顧客のデータを前記顧客別流行度及び購買タイミングデータベースから除外し、

前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの平均値またはモード値を入力し、商品のカテゴリーごとに、流行度スコアの平均値またはモード値、購買タイミングスコアの平均値またはモード値の組合せにより、流行に対する感度と商品の購買タイミングを因子とする複数の顧客類型に顧客を分類し、顧客類型データベースに出力し、

過去の商品の販売実績データを入力し、商品の売上推移の段階と、商品の各売上推移段階中に商品を購買した顧客の類型の構成比とを説明変数とし、その商品の売上げまたは流行の度合いを被説明変数として、商品のカテゴリーごとに、商品の売上予測モデルを算出し、

予測対象商品の販売データを入力し、売上推移の傾斜から商品の売上推移段階を推定し、

予測対象商品の顧客データを入力し、これらの顧客データの各顧客について前記顧客類型データベースを参照することにより、各顧客の顧客類型を特定し、

前記推定した売上推移段階と、前記特定した顧客類型と、商品カテゴリーに対応する前記売上予測モデルとを入力し、前記予測対象商品の売上げまたは流行の度合いを予測する、ことを特徴とする商品の流行予測を加味した売上予測方法。

【請求項6】商品を購買した顧客と、商品が販売された日時のデータをリアルタイムで入力し、リアルタイムで商品の売上げまたは流行の度合いを予測する、ことを特徴とする請求項4または5に記載の商品の流行予測を加味した売上予測方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、商品の売上予測を行う装置と方法に係り、特に商品の流行性の予測を加味した売上予測を行う装置と方法に関する。

【0002】

【従来の技術】従来から、商品の売上予測を行う種々の装置や方法が知られている。食品や日用品のような、購買頻度が高い商品については、予測精度がかなり高い売上予測モデルが幾つか提案されている。

【0003】これに対して、本発明が対象とするような流行性が高い商品については、従来は合理的な売上予測

モデルが提案されていなかった。

【0004】ここで、流行性が高い商品とは、購買頻度が低く、他人の目を意識したり、他人の購買によって購買行為が引き起こされたりする商品である。たとえば、アパレル商品、レジャー用商品がこれに該当する。

【0005】この流行性が高い商品に対する従来の売上予測方法は、一般的には、過去の同様の商品の販売実績データを用いて推測するものであった。すなわち、過去に販売した似通った商品の販売実績データを探し出し、それに景気動向、季節ファクター、経験値等を加味して、所定の係数を乗じたり、加えたりするものがほとんどであった。

【0006】

【発明が解決しようとする課題】しかし、上記流行性が高い商品に対する従来の売上予測方法では、合理性を欠いていた。これは、主に流行性という購買者が大きく関与するものに対して販売者側の一方的な観測を基礎にしてからである。

【0007】つまり、流行性が高い商品は、上述したように他人の目を意識したり、他人の購買によって購買行為が引き起こされたりする購買者の購買行為が大きな要素になっている。たとえば、一人の購買者が何らかの理由である最初にある商品を購買することによって、それを引き金として流行に敏感な他の購買者が購買行為に走ることがある。このことが連続的に雪崩現象的に起こることによって、商品は当初の予想をはるかに超えて売上を伸ばすことになる。これが流行、すなわちヒット商品が生まれるメカニズムである。

【0008】この場合、最初の購買者が買うか否か、また、後の購買者が追随するか否かは、商品に対する販売者側の一方的な観測では測り知ることはできなかった。

【0009】言葉を変えれば、商品の流行は、流行に敏感な購買者が商品の販売の初期段階で購買するか否かを把握することによってはじめて合理的に予測可能になる。

【0010】しかし、従来の売上予測方法は、実際に商品を販売した場合の購買者の類型と動向を加味していなかった。これでは、商品の流行予測を加味した合理的な商品の売上予測を行うことはできなかった。

【0011】これに対して部分的にはあるが、購買者の類型と動向を売上予測に加味する方法として、いわゆるモニターやアンケートを利用する方法があった。これは、ある商品について購買者から商品に対する感想等の情報をフィードバックさせ、このフィードバック情報を基に商品の売上予測を行うものであった。

【0012】しかし、この方法では、購買者とのやり取りに時間がかかってしまっていた。一般に、流行性が高い商品は、ピークを迎える時間が短いため、このように時間がかかる売上予測方法では商品の売上予測を生産販売計画に反映させることはできなかった。

【0013】一方、最近では多数の企業の顧客カード戦略によって、大量の顧客の購買実績データが蓄積されている。しかし、これらの顧客購買実績データは、結局は活用されないのが実状であった。

【0014】そこで、本発明が解決しようとする課題は、商品の販売の初期段階でその商品を購買した者の類型、構成、売上推移の動向を把握し、それによって迅速にその商品の売上予測を行う「商品の流行予測を加味した売上予測装置及びその予測方法」を提供することにある。

【0015】

【課題を解決するための手段】本願請求項1に係る「商品の流行予測を加味した売上予測装置」は、商品の品目、商品を購買した顧客、商品が販売された日時を含む商品の販売実績データを格納する販売実績データ記憶手段と、前記販売実績データ記憶手段から商品の販売実績データを検索して入力し、商品ごとに流行度指標と購買タイミング区分とを設定し、商品情報データベースに出力する流行度指標及び購買タイミング区分設定手段と、前記販売実績データ記憶手段から顧客の購買実績データを検索して入力し、顧客別に購買した商品の品目、購買時点を整理し、顧客別購買履歴データベースに出力する顧客別購買履歴作成手段と、前記顧客別購買履歴データベースから顧客別の購買履歴データを入力し、各顧客が購買した商品の全品目について前記商品情報データベースを参照し、各購買商品の流行度指標と購買時点が属する購買タイミング区分に応じてそれぞれ流行度スコアと購買タイミングスコアを付与し、前記流行度スコアと購買タイミングスコアの平均値またはモード値、標準偏差値または分散値を算出し、顧客別流行度及び購買タイミングデータベースに出力する顧客別流行度及び購買タイミングスコア算出手段と、前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの標準偏差値または分散値を入力し、前記標準偏差値または分散値により、流行度及び購買タイミングに一貫性を欠く顧客のデータを前記顧客別流行度及び購買タイミングデータベースから除外する一貫性欠如サンプル除外手段と、前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの平均値またはモード値を入力し、流行度スコアの平均値またはモード値、購買タイミングスコアの平均値またはモード値の組合せにより、流行に対する感度と商品の購買タイミングを因子とする複数の顧客類型に顧客を分類し、顧客類型データベースに出力する顧客類型化手段と、過去の商品の販売実績データを入力し、商品の売上推移の段階と、商品の各売上推移段階中に商品を購買した顧客の類型の構成比とを説明変数とし、その商品の売上げまたは流行の度合いを被説明変数として、商品の売上予測モデルを算出する売上予測モデル算出手段と、予測対象商品

について、その商品を購買した顧客と、商品が販売された日時のデータを入力する販売データ入力手段と、前記販売データ入力手段が入力した商品の販売データを入力し、売上推移の傾斜から商品の売上推移段階を推定する売上推移段階推定手段と、前記販売データ入力手段が入力した顧客データを入力し、これらの顧客データの各顧客について前記顧客類型データベースを参照することにより、各顧客の顧客類型を特定する顧客類型特定手段と、前記売上推移段階推定手段が推定した売上推移段階と、前記顧客類型特定手段が特定した顧客類型と、前記売上予測モデル算出手段が算出した売上予測モデルとを入力し、前記予測対象商品の売上げまたは流行の度合いを予測して出力する売上予測手段と、を備えたことを特徴とするものである。

【0016】本願請求項2に係る「商品の流行予測を加味した売上予測装置」は、前記請求項1の装置において、前記販売実績データ記憶手段は、商品のカテゴリーのデータを格納しており、前記顧客別購買履歴作成手段は、商品のカテゴリーごとに、各顧客が購買した商品の品目、購買時点を整理し、顧客別購買履歴データベースに出力し、前記顧客別流行度及び購買タイミングスコア算出手段は、各商品のカテゴリーごとに、前記流行度スコアと購買タイミングスコアの平均値またはモード値、標準偏差値または分散値を算出し、顧客別流行度及び購買タイミングデータベースに出力し、前記一貫性欠如サンプル除外手段は、各商品のカテゴリーごとに、流行度及び購買タイミングに一貫性を欠く顧客のデータを前記顧客別流行度及び購買タイミングデータベースから除外し、前記顧客類型化手段は、各商品のカテゴリーごとに、流行に対する感度と商品の購買タイミングを因子とする複数の顧客類型に顧客を分類し、顧客類型データベースに出力し、前記売上予測モデル算出手段は、各商品のカテゴリーごとに、商品の売上予測モデルを算出し、前記売上予測手段は、商品カテゴリーに対応する売上予測モデルを用いて前記予測対象商品の売上げまたは流行の度合いを予測する、ように構成されていることを特徴とするものである。

【0017】本願請求項3に係る「商品の流行予測を加味した売上予測装置」は、前記請求項1または2の装置において、前記販売データ入力手段は、商品を購買した顧客と、商品が販売された日時のデータをリアルタイムで入力し、前記売上予測手段は、前記販売データ入力手段がリアルタイムで入力したデータを用いて商品の売上げまたは流行の度合いを予測する、ように構成されていることを特徴とするものである。

【0018】本願請求項4に係る「商品の流行予測を加味した売上予測方法」は、商品の品目、商品を購買した顧客、商品が販売された日時を含む商品の販売実績データを格納した販売実績データ記憶手段から商品の販売実績データを検索して入力し、商品ごとに流行度指標と購買タイミング区分とを設定し、商品情報データベースに出力し、前記販売実績データ記憶手段から顧客の購買実績データを検索して入力し、商品のカテゴリーごとに、顧客別に購買した商品の品目、購買時点を整理し、顧客別購買履歴データベースに出力し、前記顧客別購買履歴データベースから顧客別の購買履歴データを入力し、商品のカテゴリーごとに、各顧客が購買した商品の全品目について前記商品情報データベースを参照

買タイミング区分とを設定し、商品情報データベースに出力し、前記販売実績データ記憶手段から顧客の購買実績データを検索して入力し、顧客別に購買した商品の品目、購買時点を整理し、顧客別購買履歴データベースに出力し、前記顧客別購買履歴データベースから顧客別の購買履歴データを入力し、各顧客が購買した商品の全品目について前記商品情報データベースを参照し、各購買商品の流行度指標と購買時点が属する購買タイミング区分に応じてそれぞれ流行度スコアと購買タイミングスコアを付与し、前記流行度スコアと購買タイミングスコアの平均値またはモード値、標準偏差値または分散値を算出し、顧客別流行度及び購買タイミングデータベースに出力し、前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの標準偏差値または分散値を入力し、前記標準偏差値または分散値により、流行度及び購買タイミングに一貫性を欠く顧客のデータを前記顧客別流行度及び購買タイミングデータベースから除外し、前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの平均値またはモード値を入力し、流行度スコアの平均値またはモード値、購買タイミングスコアの平均値またはモード値の組合せにより、流行に対する感度と商品の購買タイミングを因子とする複数の顧客類型に顧客を分類し、顧客類型データベースに出力し、過去の商品の販売実績データを入力し、商品の売上推移の段階と、商品の各売上推移段階中に商品を購買した顧客の類型の構成比とを説明変数とし、その商品の売上げまたは流行の度合いを被説明変数として、商品の売上予測モデルを算出し、予測対象商品の販売データを入力し、売上推移の傾斜から商品の売上推移段階を推定し、予測対象商品の顧客データを入力し、これらの顧客データの各顧客について前記顧客類型データベースを参照することにより、各顧客の顧客類型を特定し、前記推定した売上推移段階と、前記特定した顧客類型と、前記算出した売上予測モデルとを入力し、前記予測対象商品の売上げまたは流行の度合いを予測する、ことを特徴とするものである。

【0019】本願請求項5に係る「商品の流行予測を加味した売上予測方法」は、商品の品目、カテゴリー、商品を購買した顧客、商品が販売された日時を含む商品の販売実績データを格納した販売実績データ記憶手段から商品の販売実績データを検索して入力し、商品ごとに流行度指標と購買タイミング区分とを設定し、商品情報データベースに出力し、前記販売実績データ記憶手段から顧客の購買実績データを検索して入力し、商品のカテゴリーごとに、顧客別に購買した商品の品目、購買時点を整理し、顧客別購買履歴データベースに出力し、前記顧客別購買履歴データベースから顧客別の購買履歴データを入力し、商品のカテゴリーごとに、各顧客が購買した商品の全品目について前記商品情報データベースを参照

し、各購買商品の流行度指標と購買時点が属する購買タイミング区分に応じてそれぞれ流行度スコアと購買タイミングスコアを付与し、前記流行度スコアと購買タイミングスコアの平均値またはモード値、標準偏差値または分散値を算出し、顧客別流行度及び購買タイミングデータベースに出力し、前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの標準偏差値または分散値を入力し、商品のカテゴリーごとに、前記標準偏差値または分散値により、流行度及び購買タイミングに一貫性を欠く顧客のデータを前記顧客別流行度及び購買タイミングデータベースから除外し、前記顧客別流行度及び購買タイミングデータベースから各顧客の流行度スコアと購買タイミングスコアの平均値またはモード値を入力し、商品のカテゴリーごとに、流行度スコアの平均値またはモード値、購買タイミングスコアの平均値またはモード値の組合せにより、流行に対する感度と商品の購買タイミングを因子とする複数の顧客類型に顧客を分類し、顧客類型データベースに出力し、過去の商品の販売実績データを入力し、商品の売上推移の段階と、商品の各売上推移段階中に商品を購買した顧客の類型の構成比とを説明変数とし、その商品の売上げまたは流行の度合いを被説明変数として、商品のカテゴリーごとに、商品の売上予測モデルを算出し、予測対象商品の販売データを入力し、売上推移の傾斜から商品の売上推移段階を推定し、予測対象商品の顧客データを入力し、これらの顧客データの各顧客について前記顧客類型データベースを参照することにより、各顧客の顧客類型を特定し、前記推定した売上推移段階と、前記特定した顧客類型と、商品カテゴリーに対応する前記売上予測モデルとを入力し、前記予測対象商品の売上げまたは流行の度合いを予測することを特徴とするものである。

【0020】本願請求項6に係る「商品の流行予測を加味した売上予測方法」は、前記請求項4または5の予測方法において、商品を購買した顧客と、商品が販売された日時のデータをリアルタイムで入力し、リアルタイムで商品の売上げまたは流行の度合いを予測することを特徴とするものである。

【0021】

【発明の実施の形態】本発明による「商品の流行予測を加味した売上予測装置」は、過去の商品の販売実績データと顧客の購買実績データとから、各顧客の流行度と購買タイミングを特定し、どの売上推移段階でどのような流行指向と購買タイミングパターンの顧客が購買を行ったかということと実際の商品の売上との相関関係を求めておき、ある商品の売上を予測する場合に、その商品を購買した顧客のデータを収集し、それら顧客の流行度と購買タイミングのパターンを分析することにより、前記相関関係から商品の売上を予測しようとするものである。

【0022】ここで、各顧客の「流行度」とは、流行に敏感か否かを示す指標、すなわち流行する商品を好む度合いを示す指標である。本発明では、その顧客が過去に購買した商品の流行の程度からその顧客の流行度を特定する。具体的な方法については後にさらに詳しく説明する。

【0023】「購買タイミング」とは、商品の売上推移において、販売の初期段階、成長段階、成熟段階、衰退段階のいずれの段階で購買を行ったかということである。いずれの段階で商品を購入するかは個人によってほぼ一定していることが知られている。具体的な方法については後にさらに詳しく説明する。

【0024】次に、本発明の一実施形態について添付の図面を参照して以下に説明する。

【0025】図1に本発明の一実施形態による売上予測装置の構成と、それら構成要素間の処理の流れを示す。

【0026】図1に示すように、本実施形態による商品の売上予測装置1は、販売実績データ記憶手段2、流行度指標及び購買タイミング区分設定手段3、顧客別購買履歴作成手段4、顧客別流行度及び購買タイミングスコア算出手段5、一貫性欠如サンプル除外手段6、顧客類型化手段7、売上予測モデル算出手段8、販売データ入力手段9、売上推移段階推定手段10、顧客類型特定手段11、売上予測手段12を有している。

【0027】この他に本実施形態による商品の売上予測装置1は、上記構成手段の出力したデータを格納管理する複数のデータベースを有している。商品情報データベース13、顧客別購買履歴データベース14、顧客別流行度及び購買タイミングデータベース15、顧客類型データベース16が、これに該当する。なお、データベースとは、一般的には系統的に管理されたデータの集まりをいうが、ここではデータを格納管理するハードウェアを含めてデータベースということにする。

【0028】本明細書にいう売上予測装置1とその構成手段は、固定的にそれぞれの処理を行うようにしたハードウェアでもよい。しかし好ましくは、売上予測装置1とその構成手段は、所定のソフトウェアによって制御され、ある処理段階で処理段階に応じた処理を行うコンピュータからなる。また、売上予測装置1とその構成手段の全体が1台のコンピュータでもよいし、それぞれが独立のコンピュータからなり、それらがネットワークによって接続されて全体として売上予測装置1を構成するようにしてもよい。

【0029】販売実績データ記憶手段2は、商品の販売実績データを格納する記憶装置である。ここでいう「商品の販売実績データ」とは、商品の品目、商品のカテゴリー、商品を購買した顧客、商品が販売された日時等のデータである。販売実績データ記憶手段2は、物理的には外部の記憶装置であることがあり得る。たとえば、顧客情報や商品販売実績データを格納する外部のサーバー

であってもよい。商品の販売実績データは、データベース化されているのが好ましいが、データベース化されていないデータを売上予測装置1が取得してデータベースを構築することも可能である。従って、販売実績データ記憶手段2は、複数個あって売上予測装置1が顧客情報と商品販売情報を別々に得て、これらデータをリンクさせるようにしてもよい。

【0030】流行度指標及び購買タイミング区分設定手段3は、商品の販売実績データを入力し、商品ごとに流行度指標と購買タイミング区分とを設定する手段である。ここで、流行度指標とは、流行の程度を示す指標である。流行度指標はユーザーによって任意に定めることができる。本実施形態では、商品の（実際の売上）／（予想売上）の値によって1～5の数値を流行度指標としている。この他、人間による評価を流行度指標とすることもできる。

【0031】購買タイミング区分は、商品の売上の推移により、商品の販売の開始から実質的な終了までの時間帯を幾つかの段階に分け、そのどの段階で顧客が購買したかを特定するための区分である。商品の売上推移の段階としては、概念的には販売の初期段階、成長段階、成熟段階、衰退段階というようなものがある。具体的な、購買タイミング区分の方法は、ユーザーによって任意に定めることができる。本実施形態では、ある商品の売上推移のデータを用いて、5分位の方法によって購買タイミング区分を設定している。5分位の方法とは、購買の度数を時間順に5等分した場合の各区分である。その他、標準偏差を用いて～2シグマ、2～1シグマ、1～0シグマ、0～-1シグマ、-1シグマを5つの購買タイミング区分とすることもできる。

【0032】流行度指標及び購買タイミング区分設定手段3は、流行度指標及び購買タイミング区分設定のための入力手段を提供するものでもよいし、あるいは予め定めた基準値によって自動的に流行度指標と購買タイミングの区分を設定するものでもよい。

【0033】顧客別購買履歴作成手段4は、顧客の購買実績データを入力し、顧客別に購買した商品の品目、商品のカテゴリー、購買時点を整理して出力するものである。

【0034】顧客別流行度及び購買タイミングスコア算出手段5は、顧客別の購買履歴データを入力し、各顧客が購買した商品の全品目について前記流行度指標と購買タイミング区分を参照し、それぞれの流行度指標と購買タイミング区分に応じて得点を付与し、これら得点の平均値、モード値、標準偏差値、分散値等を算出する手段である。

【0035】ここで、流行度指標と購買タイミング区分に応じて付与する得点が、その顧客の「流行度スコア」と「購買タイミングスコア」である。つまり、「流行度スコア」と「購買タイミングスコア」は、購買し

た商品のデータに基づいて顧客の流行指向と購買タイミングパターンを数値化したものである。

【0036】一貫性欠如サンプル除外手段6は、顧客の流行度スコアと購買タイミングスコアの標準偏差値または分散値を入力し、標準偏差値または分散値により、流行度及び購買タイミングに一貫性を欠く顧客のデータを売上予測用のデータから除外する手段である。これら購買行動に一貫性を欠く顧客のデータは、商品の売上予測をするための基礎データとしては不適だからである。なお、標準偏差値や分散値の他にサンプルのばらつきを示す任意の指標を使用することができる。

【0037】顧客類型化手段7は、顧客の流行度スコアと購買タイミングスコアの平均値またはモード値を入力し、流行度スコアの平均値またはモード値、購買タイミングスコアの平均値またはモード値の組合せにより、流行に対する感度と商品の購買タイミングを因子とする複数の顧客類型に顧客を分類する手段である。具体的な分類方法については後にさらに説明する。

【0038】売上予測モデル算出手段8は、過去の商品の販売実績データを入力し、商品の売上推移の段階と、商品の各売上推移段階中に商品を購買した顧客の類型の構成比とを説明変数とし、その商品の売上げまたは流行の度合いを被説明変数として、商品の売上予測モデルを算出する手段である。なお、本実施形態による売上予測モデルの算出は回帰分析の方法を採用しているが、売上予測モデルの算出は、適当な関数を発見することによる方法、関数の形を特定できなくても一定の入力に対して一定の出力を得る方法（たとえばニューラルネットワーク）、その他任意の公知の方法を採用することができる。

【0039】販売データ入力手段9は、予測対象商品について、その商品を購買した顧客と、商品が販売された日時のデータを入力する手段である。販売データ入力手段9は、販売データをリアルタイムで収集することができる手段であることが好ましい。

【0040】売上推移段階推定手段10は、売上を予測しようとする商品の販売データを入力し、その売上推移の傾斜から商品の売上推移段階を推定する手段である。

【0041】顧客類型特定手段11は、売上を予測しようとする商品を買った顧客のデータを入力し、各顧客について流行性指向と購買タイミングパターンの類型を特定する手段である。流行性指向と購買タイミングパターンの類型の特定方法は後にさらに説明する。

【0042】売上予測手段12は、売上を予測しようとする商品の売上推移段階と、それを購買した顧客の類型と、売上予測モデルとを入力し、予測対象商品の売上げまたは流行の度合いを予測する手段である。

【0043】以上が本実施形態の売上予測装置1の各構成手段についての説明を終了する。次に、上記構成手段による売上予測装置1の「商品の流行予測を加味した売

上予測」の方法について図1に沿って説明する。

【0044】最初に、過去の販売実績データを用いて、商品の売上予測を行うための基礎データの収集と作成を行う。

【0045】まず、流行度指標及び購買タイミング区分設定手段3により、販売実績データ記憶手段2から商品の販売実績データを検索し、商品ごとに流行度指標と購買タイミング区分とを設定する。図2に流行度指標と購買タイミング区分の設定の方法を示す。

【0046】図2に示すように、販売実績データ記憶手段2は、顧客、商品カテゴリー、商品品目、販売日時、等のデータを有している。流行度指標及び購買タイミング区分設定手段3は、販売実績データ記憶手段2から商品の販売実績データを抽出する。商品の販売実績データは、時間を横軸、売上（度数でもよい）を縦軸とする。図2に示すような売上推移カーブで表わされる。この売上推移カーブについて、図2に示すような購買タイミング区分 t_{a1}, \dots, t_{a5} を設定することができる。また、実際の売上) / (予想売上) の値を用いて流行度指標(1~5)を設定することができる。これにより、図2の下部にしめすような商品情報を作成することができる。流行度指標及び購買タイミング区分設定手段3により、商品情報を商品情報データベース13に出力する。

【0047】次に、顧客別購買履歴作成手段4により、顧客別購買履歴を作成する。

【0048】顧客別購買履歴作成手段4は、販売実績データ記憶手段2から顧客の購買実績データを検索し、顧客別に購買した商品の品目、購買時点を整理して顧客別の購買履歴を作成する。図3に販売実績データから顧客別購買履歴データを作成する方法を示す。

【0049】既に説明したように、販売実績データ記憶手段2は、顧客、商品カテゴリー、商品品目、販売日時、等のデータを有している。顧客別購買履歴作成手段4は、顧客の購買実績データを抽出し、図3に示すように顧客CL1, CL2, CL3, …ごとに、かつ、商品カテゴリーA, B, …ごとに、購買した商品品目と購買した日時(図3においてタイミングの欄)をリストアップする。これが「顧客別購買履歴データ」である。顧客別購買履歴作成手段4は顧客別購買履歴データを顧客別購買履歴データベース14に出力する。

【0050】次に、顧客別流行度及び購買タイミングスコア算出手段5により、顧客別の流行度スコアと購買タイミングスコアとを算出する。

【0051】顧客別流行度及び購買タイミングスコア算出手段5は、顧客別購買履歴データベース14から顧客別の購買履歴データを入力し、各顧客が購買した商品の全品目について商品情報データベース14を参照し、各購買商品の流行度指標と購買時点が属する購買タイミング区分に応じてそれぞれ流行度スコアと購買タイミ

ングスコアを付与する。流行度スコアと購買タイミングスコアの付与の方法を図4に示す。

【0052】図4に示すように、顧客別購買履歴データは、顧客が購買した商品と、各商品○○○, □□□, …を購買した日時d/m/yのデータを有している。顧客別流行度及び購買タイミングスコア算出手段5は、各商品について、商品情報データベース13を検索し、該当する商品の購買タイミング区分tn1, tn2, …と、流行度指標(1~5)を参照し、該当する購買タイミング区分tn1, tn2, …と流行度指標(1~5)にそれぞれ対応した購買タイミングスコアと流行度スコアを付与する。さらに、顧客別流行度及び購買タイミングスコア算出手段5は、商品カテゴリーA, B, …ごとに、購買タイミングスコアと流行度スコアの平均値と標準偏差値を算出する。

【0053】なお、商品カテゴリーA, B, …ごとに購買タイミングスコアと流行度スコアを付与するのは、一般に同一の顧客であっても、商品カテゴリーAについては流行度が高いが、商品カテゴリーBについては低い、というようなことがしばしば起きるからである。このように、商品カテゴリーA, B, …ごとに購買タイミングスコアと流行度スコアを付与することにより、より正確な売上予測が可能になる。

【0054】顧客別流行度及び購買タイミングスコア算出手段5は、算出した「顧客別流行度及び購買タイミングデータ」を顧客別流行度及び購買タイミングデータベース15に出力する。

【0055】次に、一貫性欠如サンプル除外手段6により、流行指向と購買タイミングパターンに一貫性がない顧客データを顧客別流行度及び購買タイミングデータベース15から除外する。

【0056】一貫性がない顧客データを除外するには、標準偏差値が一定の閾値より大きいものを除外する。一貫性欠如サンプル除外手段6による一貫性がない顧客データの除外は、カテゴリーごとに、また、流行度または購買タイミングごとに行うことができる。

【0057】次に、顧客類型化手段7により、顧客を流行指向と購買タイミングパターンに応じて類型化する。

【0058】顧客類型化手段7は、一貫性のないデータを除外した顧客別流行度及び購買タイミングデータベース15から、各顧客の流行度スコアと購買タイミングスコアの平均値を入力し、それらの組合せにより、流行度と商品の購買タイミングを因子とする複数の顧客類型に顧客データを分類する。図5に類型分類の方法を示す。

【0059】図5に示すように、顧客の類型分類を行うには、類型表を使用する。この類型表は購買タイミングスコア平均値(PTS)の所定の帯域と流行度スコア平均値(HTS)の所定の帯域とのマトリックスからなる。この類型表を参照することにより、各顧客の商品力

テゴリーごとの類型イ、ロ、ハ、ニ、ホ、…を特定することができる。なお、図5の顧客類型データにおいて、商品カテゴリーにより類型が特定されていないものがあるのは、その顧客がその商品カテゴリーについては流行に関心がないか、あるいは購買に一貫性がないことを示している。

【0060】顧客類型化手段7は、上記のように作成した顧客類型データを顧客類型データベース16に出力する。

【0061】次に、売上予測モデル算出手段8により売上予測モデルを算出する。売上予測モデル算出手段8は、過去の商品の販売実績データ（図2参照）を入力し、商品の売上推移の段階と、商品の各売上推移段階中にその商品を購買した顧客の類型の構成比とを説明変数とし、その商品の売上げまたは流行の度合いを被説明変数として、商品の売上予測モデルを算出する。商品の売上予測モデルの算出は、すでに説明したように、重回帰分析、所定の関数の設定、ニューラルネットワークのいずれの方法でもよい。

【0062】なお、商品の売上予測モデルは原則的には商品カテゴリーごとに算出するが、これに限られず、各商品ごとあるいは一定のカテゴリーを統合した商品群について算出してもよい。

【0063】以上が過去の販売実績データを用いて、商品の売上予測を行うための基礎データの収集と作成である。なお、上記説明では、平均値と標準偏差値を用いて説明したが、平均値の代わりにモード値、標準偏差値の代わりに分散を使用してもよいことは説明するまでもない。

【0064】次に、上記売上予測モデルを用いた商品の売上予測について説明する。

【0065】本発明の方法によりある商品の売上予測を行うには、その商品の販売データを収集しなければならない。

【0066】販売データ入力手段9は、好ましくはリアルタイムで商品の販売データを売上予測装置1に送るものとする。

【0067】次に、売上推移段階推定手段10により、その商品の売上推移段階を推定する。商品の売上推移段階を推定するには、商品の販売データから売上推移の傾斜を求め、傾斜の大きさ、あるいは傾斜の増減等により売上推移段階を推定することができる。また、過去の販売経験から判断して商品サイクルがほぼ固定的な特性を示すものであると仮定できる場合は、その時点における販売データから直ちに売上推移段階を決定することもできる。

【0068】次に、顧客類型特定手段11により、その商品を購入している顧客の類型を特定する。顧客類型特定手段11は、販売データ入力手段9が入力した顧客データを入力し、各顧客について顧客類型データベース1

6を検索し、該当する顧客データからその顧客類型を特定する。

【0069】このように、売上を予測しようとする商品の売上推移段階と購買している顧客の類型を特定できれば、売上予測モデルを用いて売上または流行の度合いを算出することができる。

【0070】売上予測手段12は、売上推移段階推定手段10が推定した売上推移段階と、顧客類型特定手段11が特定した顧客類型と、売上予測モデル算出手段8が算出した売上予測モデルとを入力し、それらのデータと数式等を用いて予測対象商品の売上げまたは流行の度合いを予測することができる。

【0071】このように、本発明の売上予測装置によれば、過去の販売実績データから顧客の流行指向と購買タイミングパターンを特定し、商品のある販売推移段階において購買している者の流行指向と購買タイミングパターンと商品の売上の関係を売上予測モデル化している。これにより、ヒット商品が生まれるメカニズムに沿った商品の売上予測を行うことができる。

【0072】また、上記売上予測装置1によれば、売上予測を行うコンピュータ（売上予測手段12）を常時起動しておき、それに商品の販売データをリアルタイムでフィードバックすれば、常にリアルタイムで商品の売上予測を行うことができる。これにより、モニターやアンケートの場合のような通信のための時間が省かれ、迅速に商品の流行を察知して生産販売計画に反映させることができる。

【0073】なお、上記実施形態は、商品カテゴリーごとに、顧客の流行指向、購買タイミングパターン、売上予測モデルを算出し、これに基づいて特定の商品の売上を予測するものであった（本願請求項2、5に対応）。しかし、商品カテゴリーごとに限られず、ある商品について販売実績データの蓄積があれば、同様に売上予測モデルを算出し、これに基づいて特定の商品の売上を予測することもできる。本願請求項1、4はこれに対応するものである。

【0074】また、上記実施形態では、売上を予測しようとする商品の販売データをリアルタイムで収集するようになっていたが、リアルタイムで商品の売上予測を行うことが必要条件でないことは説明するまでもない。

【0075】

【発明の効果】以上の説明から明らかなように、本発明による「商品の流行予測を加味した売上予測装置及びその予測方法」によれば、売上を予測しようとする商品を実際に購入している者の流行指向や購買タイミングパターンを分析し、これによって流行の現象に大きく関係する流行感度が高い購買層の動向を把握し、これによって商品の流行を予測することができる。この予測方法は、流行のメカニズムによく沿い、これによって合理的な商品の流行予測を加味した売上予測装置及びその予測方法

を提供することができる。

【0076】また、本発明の予測装置及びその予測方法によれば、商品の流行を販売極めて早い段階で予測でき、商品の生産販売計画に迅速に予測結果を反映させることができる予測装置及びその予測方法を提供することができる。

【図面の簡単な説明】

【図1】本発明の一実施形態による「商品の流行予測を加味した売上予測装置」の構成とその処理の流れを示したブロック図。

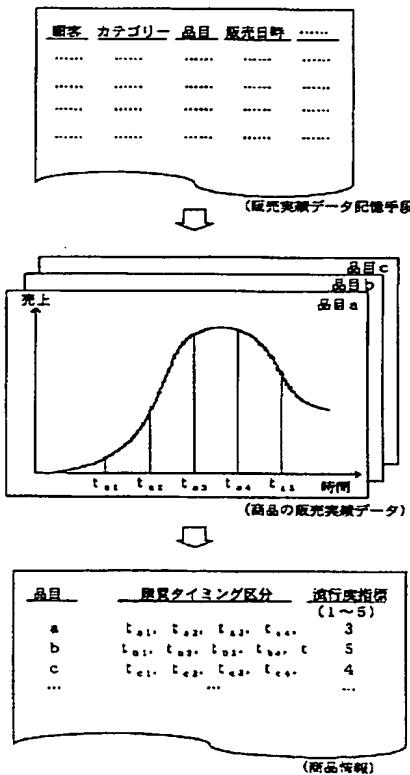
【図2】商品の販売実績データから流行度指標と販売タイミング区分を設定する方法を説明した図。

【図3】商品の販売実績データから顧客別の購買履歴データを作成する方法を説明した図。

【図4】顧客別購買履歴データと商品情報から顧客別の流行感度スコア及び購買タイミングスコアを算出する方法を説明した図。

【図5】顧客別の流行感度スコア及び購買タイミングスコアから各顧客の流行指向と購買タイミングパターンの

【图2】

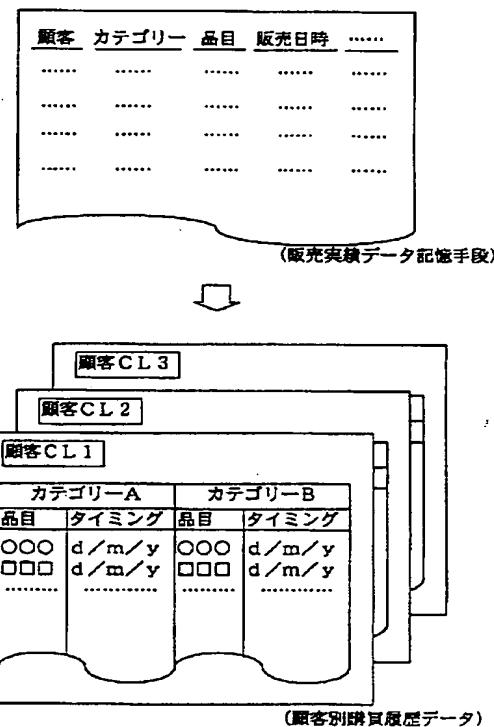


類型を特定する方法を説明した図。

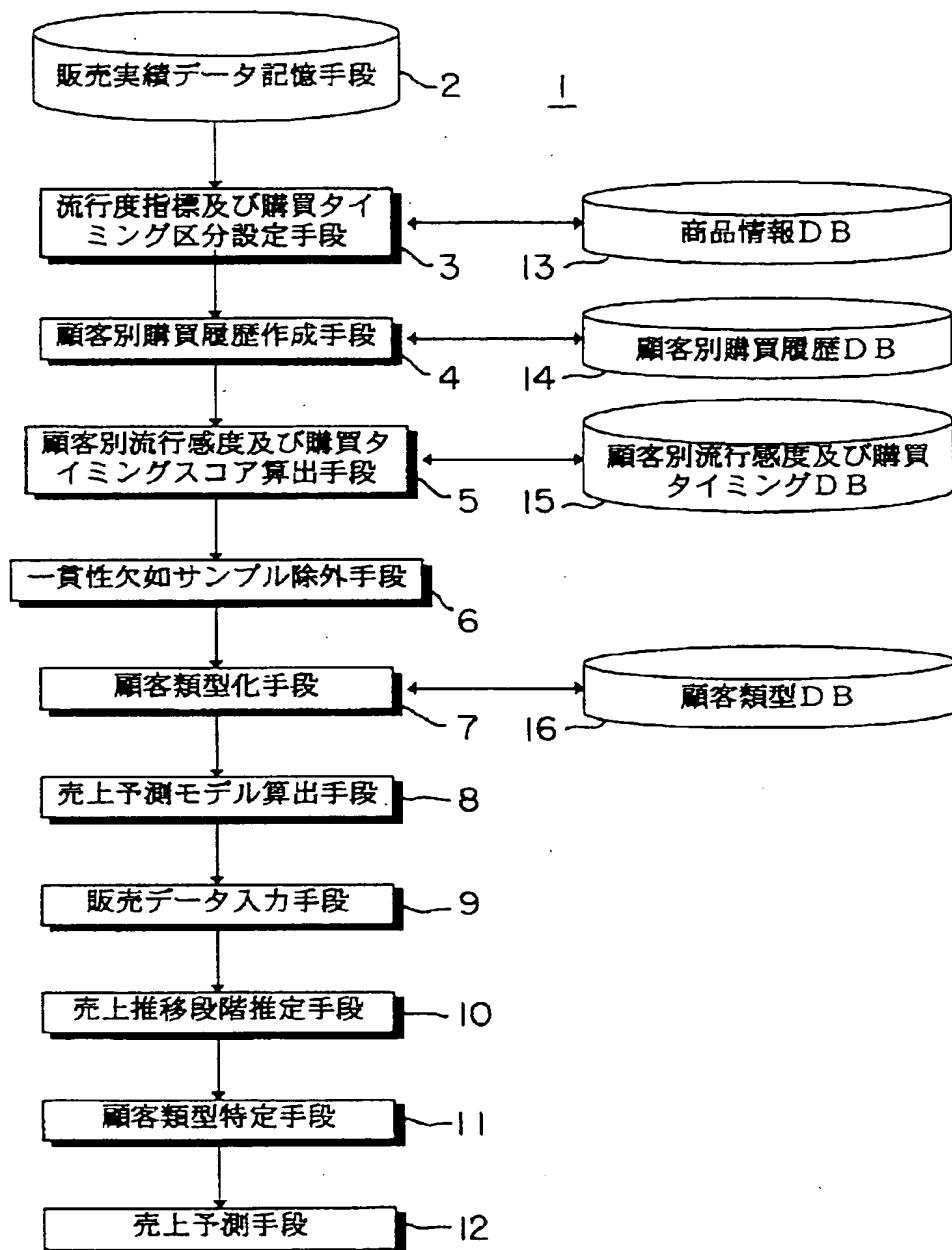
【符号の説明】

- 1 売上予測装置
 - 2 販売実績データ記憶手段
 - 3 流行度指標及び購買タイミング区分設定手段
 - 4 顧客別購買履歴作成手段
 - 5 顧客別流行感度及び購買タイミングスコア算出手段
 - 6 一貫性欠如サンプル除外手段
 - 7 顧客類型化手段
 - 8 売上予測モデル算出手段
 - 9 販売データ入力手段
 - 10 売上推移段階推定手段
 - 11 顧客類型特定手段
 - 12 売上予測手段
 - 13 商品情報データベース
 - 14 顧客別購買履歴データベース
 - 15 顧客別流行感度及び購買タイミングデータベース
 - 16 顧客類型データベース

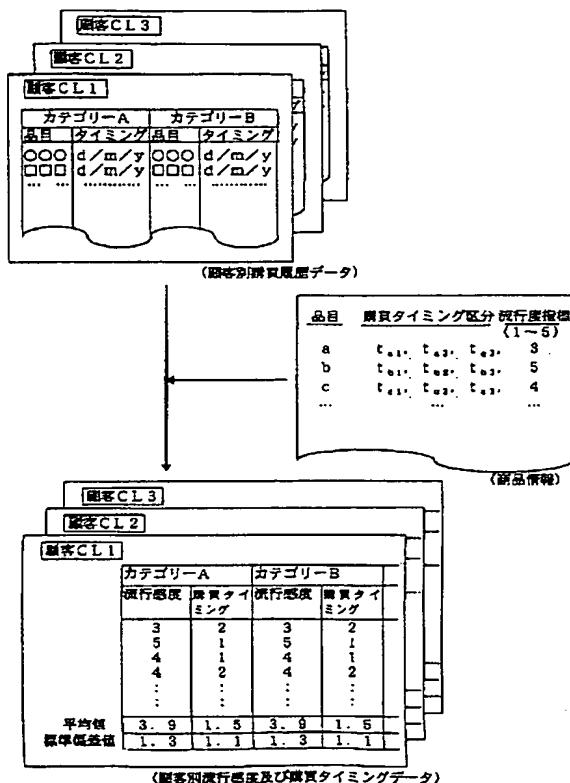
[図3]



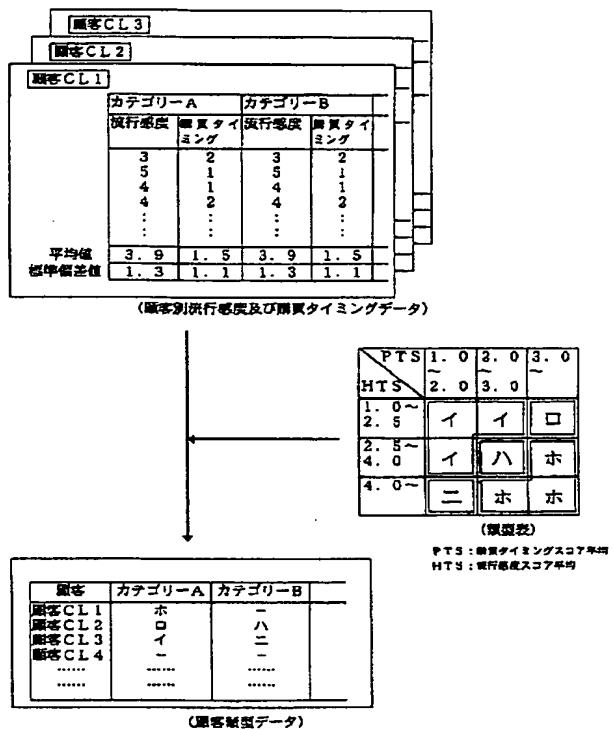
[図1]



[⊗ 4]



[図5]



CRE~~E~~T SCORING (partial translation)

E5653

II. 2. 9 Score Point Calculation

After attribute items of characteristics are classified, a score table is to be prepared in a next step. The table shows scores of respective attributes assigned to the pertinent applicant. An example of the score table is shown in Table 1. A total of scores set to appropriate attributes is "credit score" which usually called "score".

Table 1 Application scoring example

a-1	Tenure of current office	a-2	6 months or less		
a-3	6 months - 1 year 6 months				
a-4	1 year 7 months - 6 years 8 months				
a-5	6 year 9 months - 10 years 5 months				
a-6	10 years 6 months or more				
b-1	Own house or rented house				
b-2	Own house or own house to be purchased				
b-3	Rented house	b-4	Others		
c-1	Bank account	c-2	Current account		
c-3	Ordinary account	c-4	Current and ordinary accounts		
c-5	None				
d-1	Primary credit card	d-2	With credit card		
d-3	Without credit card				
e-1	Occupation	e-2	Retired		
e-3	Specialist	e-4	Office worker		
e-5	Sales division member				
e-6	Service division member				
e-7	Others				
f-1	Applicant age	f-2	18 - 25	f-3	26 - 31
f-4	32 - 34	f-5	35 - 51	f-6	52 - 61
f-7	62 or more				
g-1	Worst credit query record				
g-2	Essential credit lost	g-3	Not-serious credit lost		
g-4	No records	g-5	Good result 1		
g-6	Good result 2 or more	g-7	No results checked		

空欄の処理

偶然である、意図的である、申込者がクレジット申込書の質問に答えるないとき、申込書はスコアリング・システム開発者に特別な問題を提起する。これは特に郵送する申込書に起ることがある。

空欄は2通りの処理方法がある。予測可能情報を含むとみるか、何も含まないとみるかのどちらかである。

空欄はなんら予測可能な情報を含まないとみることにすれば、空欄は予測可能な情報を含むその他のデータから分離するべきだ。その時点からスコア表を計算する全過程で「空欄」属性項目は何も予測力がないことを示さざるをえない。これはつまり、申込者が空欄のままにしておいた質問には、スコアリング表ユーザーは特性からなんの結論も引き出せないということである。要するに、その申込者に関してはその特性はスコアリング表の一部でないかのように取り扱われる。

しかし、ユーザーが空欄から結論を出すことを選択する例もある。たとえば、クレジットカードの郵送申込書の場合、ユーザーは属性の空欄属性項目が情報を含んでいるかどうかるために、調べることにするかもしれない。「よい口座」申込者の80%が、一方「不良口座」申込者の10%が、ある特定の属性項目を空欄のまま残していくなら、ユーザーは空欄が有効な属性項目とみなすことを決定し、スコアリング表に入れることを認めるかもしれない。

II.2.9 得点 (Score Point) の計算

特性の属性項目が分類されると、次の手順はスコア表の準備である。これはそれぞれの属性項目につき、申込者に与えた得点数を示す表である。

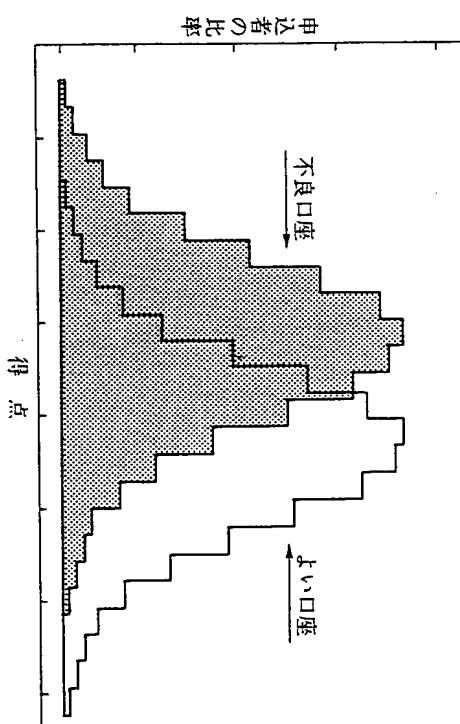
スコア表の例は序章の図表1および図表29-31に示している。それでは特性的適当な属性項目について与えた得点の合計が、「クレジット・スコア」

であり、通常単に「スコア」と呼ばれる。

これは多变量統計に関する論文ではないので、分類されたデータから得点が計算されるいろいろな方法を解説するつもりはない。この計算にさまざまな行程があり、どれもかなり有効である。どのような方法が使われようとも、よい口座と不良口座の集団をスコア表の得点を使って計算したとき、よい口座、不良口座ともに得点の低い場合と、高得点の場合があることがわかるだろう。しかし、そのスコアリング・システムに予測能力があるとすれば、よい口座のほうが高得点が多く、不良口座のほうが低得点が多いだろう(注21)。

図表21はそれぞれのスコアに入る部分の数をグラフにするとどうなるかを示している。

図表22は同じ情報を示すが、滑らかな曲線にしている。よい口座数、不良口座数とともに平均値近辺に分布していることがわかる。この例では、よい口座(注21) 得点で層別したよい口座比率、不良口座比率

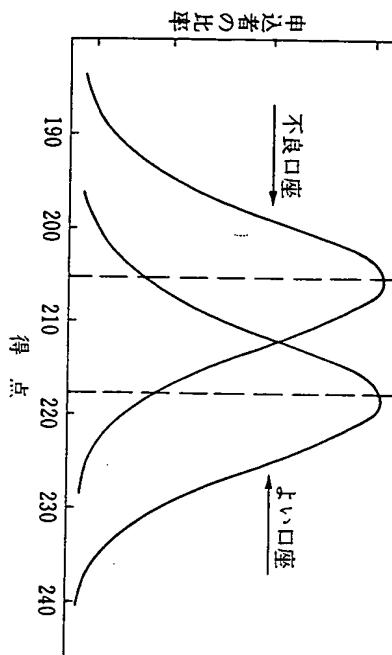


(注21) クレジット・スコアリングの分野のはとんどでは高得点が低いリスク(高い見込み(オッズ)はよい口座)を示し、低得点が高いリスクを示すようにスコア表を作成することが慣習となっている。これはまったくの恣意的選択である。

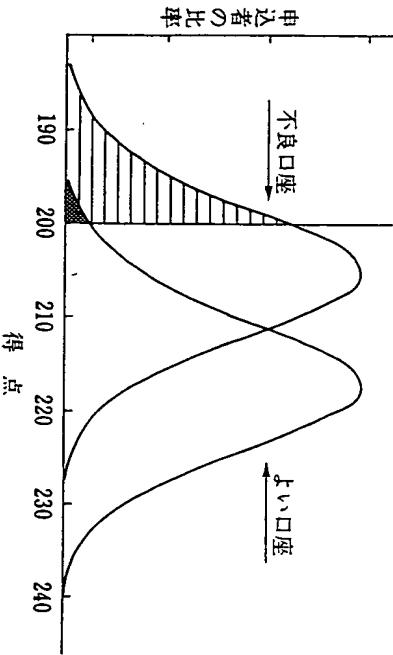
い口座は218点付近、不良口座は205点付近に寄っている。この図表はまた、高い得点をもつ不良口座もあれば、得点の低いよい口座があることも示している。これはクレジット担当者の経験を数値的に反映したものである。申込時にリスク面で優れていると思われた口座が不良になることがあるかと思えば、一方、危険だと思われた申込みが模範的に変るものもある。

図表23は最低限の許容得点を設定したときの影響を示している。200点のところで線を引いている。それ以上の得点の申込書だけを承認すれば、そ

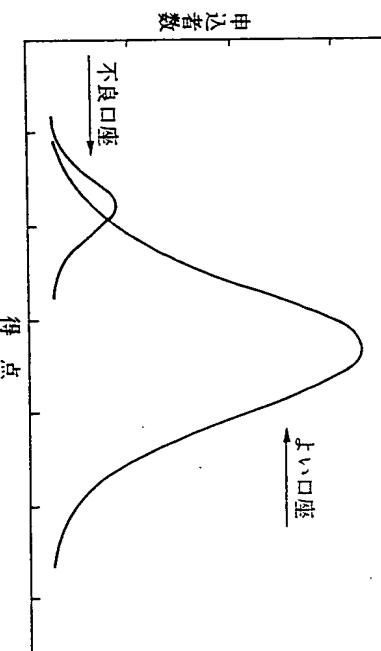
(図表22) よい口座・不良口座比率——平滑化したデータ



(図表23) カットオフ点の与える影響の例



(図表24) 実数口座の平滑分布



どちらの曲線も実際、両方向に無限に減少しているので不良口座にならないと、保証できる得点はない。

この曲線はよい口座、不良口座の実数ではなく、単に比率を表わしているにすぎない。実際には、不良口座数はよい口座数よりもはるかに少なく、不良口座、よい口座の実数グラフは図表24の曲線のようになる。

リスクを予測する二つのスコアリング・システムの相対的能力を図表25に示す。どちらの曲線もよい口座、不良口座数が得点に層別され滑らかに分散している。カットオフ点を横軸のどこかに設定すると図表のとおり、最低点ラインで、上段の二つの曲線は不良口座のおよそ半数が承認されることを示している。下段の二つの曲線の場合には不良口座の4分の1だけ

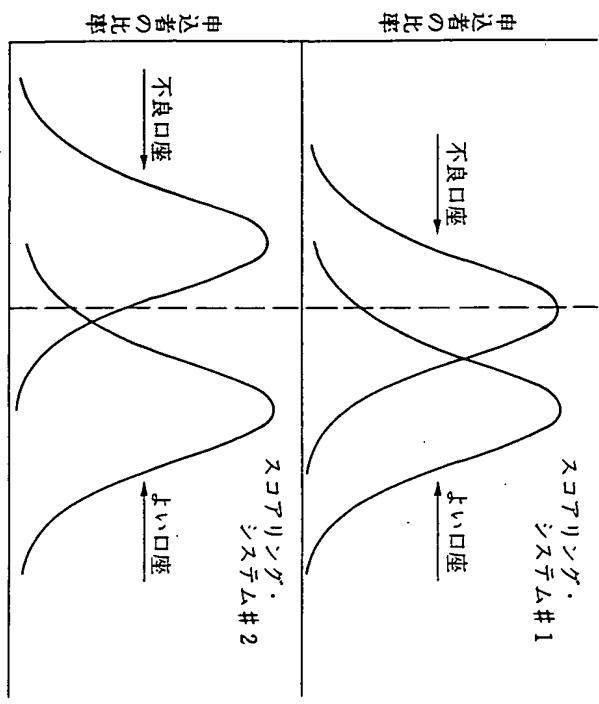
のスコアリング・システムを使用している企業は線の左側にある横線の部分の不良口座集団を回避していることがわかるが、線の左側の黒い部分のよい口座の小さな集団の承認を見合わせることになる。最低点ラインを右へ移動させるとさらに多くの不良口座が切り捨てるが、よい口座もより多く見合わせることになり、最低点ラインを左へ移動させるとその反対の事態を招く。最低点ラインが設定されている得点はカットオフ点と呼び、II.3.2でさらに詳しく解説する。

が承認されることになるが、二つの例のどちらも承認されるよい口座の数は同じである。したがって、下段の二つの曲線をつくりだしたシステムのスコアリング表は上段より能力があり、よい口座と不良口座を分離する効果がある。

スコアリング表のパフォーマンスを測るこの力は、クレジット・スコアリングの重要な特徴の一つである。二つのスコアリング・システムを使ってよい口座、不良口座集団の得点をとり、それぞれの平均点の間の距離を測ることで、二つのスコアリング・システムを比較することができる。分離度が大きいほうが、パフォーマンスが優れている。

増大 (Augmentation) —— 不承認推論

これまで、よい口座、不良口座だけを話題にし、不承認の申込書について

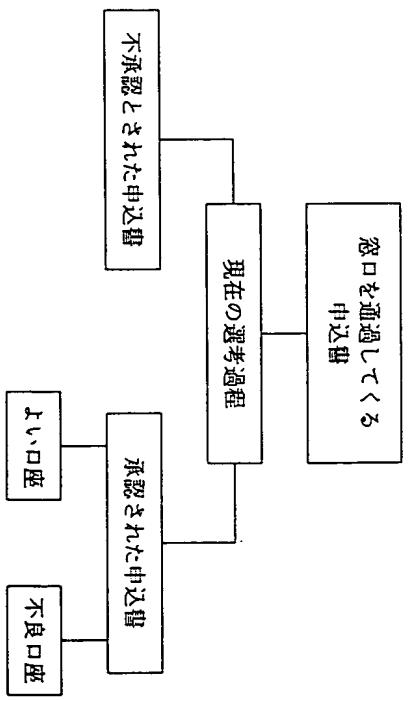


(図表25) ニつのスコアリング・システムの性能の比較

目標としては、現在の評価方法ではなく、伝統的な方法で承認される集團以外の申込集団にスコアリング・システムを適用することである。したがって、承認された申込者の情報だけでなく、窓口を通過してくる全部の情報を基礎とすることが必要である。

不承認となつた申込者が承認されいたら、どのような実績をあげていたかを推論することが必要である。申込不承認者にはその機会は与えなかつたので、どのような返済実績をあげるかはっきりと知ることは不可能である。

(図表26) 申込書の通常の流れ



ではほとんど注意を払わなかつた。申込書の流れは図表26に示されている。申込書のうち、何割かは不承認となり、承認された申込者が最終的によい口座、不良口座に分けられる。この割合は企業や、その方針により、低い数値(5~10%)から高いもの(60~75%)までさまざまである。

序 章

現在「クレジット・スコアリング」として一般的に理解されているものは、クレジット申込者やクレジット口座に関する情報が数字に置き換えられ、組み合わせて(通常は加算されて)、点数付けされる過程である。この点数は、借り入れを申し込む個人の信用力を示す(注)。

図表1は、クレジット・スコアリングの表とはどのようなものかという例である。

本書の目的は、なぜクレジット・スコアリングが生まれたか、どのように

(図表1) 申込スコアリング表の例

現在の仕事に就いている年数	6ヶ月未満	6ヶ月～1年6ヶ月	1年7ヶ月～6年8ヶ月	6年9ヶ月～10年5ヶ月	10年6ヶ月以上
C-1	a-2 5	a-3 14	a-4 2	c-5 27	a-6 39
持家あるいは賃貸	持家あるいは賃貸購入予定	質 貸	その他		
b-1	b-2 40	b-3 19	b-4 26		
所有銀行口座	当座預金	普通預金	当座預金 お預り 普通預金	なし	
C-1	c-2 22	c-3 17	c-4 31	c-5 0	
主要クレジットカード	所 持	所持せず			
d-1	d-1-2 27	d-1-3 11			
職 業	引 退	専門職	専業職	営業職	サービス業 その他
e-1	e-2 41	e-3 36	e-4 27	e-5 18	e-6 12 e-7 27
申請者の年齢	18歳～25歳	26歳～31歳	32歳～34歳	35歳～51歳	52歳～61歳 62歳以上
f-1	f-2 19	f-3 14	f-4 22	f-5 26	f-6 34 f-7 40
最悪の信用照会記録	重要な信用失墜	深刻でない 借入失墜	記録なし	優良実績 1 2以上	調査実績 なし
g-1	g-2 -15	g-3 -4	g-4 -2	g-5 9	g-6 18 g-7 0

(注1) 誰でも「信用力」という言葉の意味はわかると思うので、この言葉を使う。しかし実際では、人により考え方違うので、これについてのちはどう触れたい。いまのところは、借入申込者のクレジット・リスクということにしておこう。

かった情報が、制限から省かれた特性のなかになかったら、結果としてのスコアリング表の威力は減少する。法律の制限がある年齢以外には、制限がスコアリング・システムの弱体化を招くような負担が必要かどうか経営陣が決断できるように、制限がある場合とない場合のスコア表とを比較したものを持つことが望ましい。

個人信用情報センターの情報

これまで、通常な特性については、すべて同時に考慮できるかのように得点計算を論じてきた。その方法を支持する意見も多いが、もう一つの選択肢を支持する論拠もある。個人信用情報センターの情報はクレジット・リスクを決定するのにきわめて有用なことはわかっているが、費用がかかり、多くの場合、レポートが手に入らない。

個人信用情報センター・レポートの費用と、そのレポート欠落の可能性という観点から、得点計算プログラムを二つのスコア表を作成するようにならなければならない。その特徴は申込書の特性の得点だけのものである。そして2番目は申込書の得点に含まれていない、「個人信用情報センター」の特性の予測情報得点を含むものである。二つに変更できれば、この情報で決定が下せるかどうか、まず、申込データだけで得点をつける。申込書情報の得点がきわめて低ければ、個人信用情報センター・レポートの得点がどんなによくても、最終得点はカットオフ点を超えないだろう。その場合、その申込書は個人信用情報センターに報告を受ける時間と費用をかけず不承認にできるのである。

得点（スコア）表

わたしがよく知っている得点計算プログラムはプラスにもマイナスにもなる得点をそれぞれの特性について計算している。図表29は初期段階の成果がどのようなものかを示す架空例である。どの特性にも、「空欄」という

属性項目がある。得点は0点となっている。「最悪の信用照会記録」の特性では、「記録なし」の属性項目である。

この得点表をコンピュータ・プログラムに入れるつもりならば、このままよい。なぜなら、コンピュータは、きちんとプログラムされれば、間違いない計算を実行するからである。しかしこの表を手作業の得点計算に使うならば、できるだけマイナス記号を除いたほうがよい。なぜなら加算、減算の混合では間違いが起こるからである。図表30はどの得点も0か、プラスになるように、それぞれの特性の属性項目の得点を加算した結果を示している。最大のマイナス値を0にまで引き上げる得点をそれぞれの属性（「最悪の信用照会記録」以外）に加えるため、図表29の得点に全部で53点加算されている。どの得点もプラスになると空白とその他の明白な誤りを

(図表29) 申込スコアリング表の例

現在の仕事に就いている年数	6ヶ月未満	6ヶ月～1年6ヶ月	1年7ヶ月～6年8ヶ月	6年9ヶ月～10年5ヶ月	10年6ヶ月～10年以上	空欄
持家あるいは賃貸	-9	0	6	13	25	0
持家あるいは賃貸予定	15	-5	2	0		
所有銀行口座	5	0	14	-17	-0	
主要クレジットカード	10	-6	0			
職業	21	16	7	-2	-8	7
申込者の年齢	18歳～25歳	26歳～31歳	32歳～34歳	35歳～51歳	52歳～61歳	62歳以上
最高の信用照会記録	-3	-8	0	4	12	18
重要な信用情報失墜	-15	-4	-2	9	18	0

組み入れていた「空欄」属性項目はもはや0ではない。

「個人信用情報センター」特性（最悪の信用照会記録）の属性項目の得点はそのままマイナスである。ユーザーのなかには個人信用情報センターの情報に関する情報を含めすべての数値がプラスなことを好むかもしれません。しかし、わたしの経験では多くのユーザーは信用失墜情報にはマイナス点を使うことを選んでいた。

すべての点をプラスに変換して、かつ、カットオフ点を200点付近に設定したがるユーザーもいる。II.3.2で述べるように、申込者のたとえば、60%を承認するようカットオフ点が設定され、それが図表30の得点から計算して159点であることがわかった場合、表の全体に41点を分散すれば、カットオフ点は200点に移動するだろう。図表31はこの手順の結果を示す。「主

(図表30) どの得点もプラスもしくは0になっている申込スコアリング表の例

現在の仕事に就いている年数	6ヶ月未満	6ヶ月～1年6ヶ月	1年7ヶ月～6年8ヶ月	6年9ヶ月～10年5ヶ月	10年6ヶ月以上	空欄
持家あるいは賃貸借	0	9	15	22	34	9
持家あるいは賃貸借予定	20	0	7	5		
所有銀行口座	当座預金	普通預金	当座預金および普通預金	なし	空欄	
主要クレジットカード	22	17	31	0	17	
職業	引退	専門職	事務職	営業職	サービス業	その他
申込者の年齢	18歳～25歳	26歳～31歳	32歳～34歳	35歳～51歳	52歳～61歳	62歳以上
最悪の信用照会記録	12	7	15	19	27	33
重要な信用失墜記録	-15	-4	-2	9	18	0
最悪の信用失墜記録	5	0	8	12	20	26
申込者の年齢	18歳～25歳	26歳～31歳	32歳～34歳	35歳～51歳	52歳～61歳	62歳以上
重要なお問い合わせ記録	-15	-4	-2	9	18	0

要クレジットカード」特性ではそれぞれの属性項目に6点が加算され、「最悪の信用照会記録」以外のすべての特性の属性項目には7点が加算されている。

一部、もしくは全部の特性の全属性項目に得点を加算すると、どの申込者の得点も変化するが、得点つけされた申込者集團の格付順位は変化ない。なぜなら全員に同じ加算点を加えるからである。

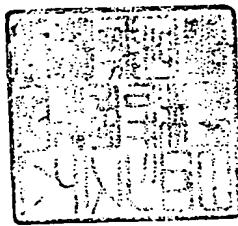
(図表31) 追加得点を加えて、すべての得点をプラスもしくは0とした申込スコアリング表の例

現在の仕事に就いている年数	6ヶ月未満	6ヶ月～1年6ヶ月	1年7ヶ月～6年8ヶ月	6年9ヶ月～10年5ヶ月	10年6ヶ月～10年5ヶ月	10年6ヶ月以上	空欄
持家あるいは賃貸借	7	16	22	29	41	16	
持家あるいは賃貸借予定	27	7	14	12			
所有銀行口座	当座預金	普通預金	当座預金および普通預金	なし	空欄		
主要クレジットカード	29	24	38	7	24		
職業	引退	専門職	事務職	営業職	サービス業	その他	空欄
申込者の年齢	36	31	22	13	7	22	15
最悪の信用照会記録	12	7	15	19	27	33	15
重要な信用失墜記録	-15	-4	-2	9	18	0	
申込者の年齢	18歳～25歳	26歳～31歳	32歳～34歳	35歳～51歳	52歳～61歳	62歳以上	空欄
重要なお問い合わせ記録	-15	-4	-2	9	18	0	

CREDIT SCORING

クレジット・スコアリング 入門

E.M.ルイス 著
木下 恭輔 監修
アコム・プロジェクト・チーム 訳



株式会社アコム
金融財政事情研究会

THIS PAGE BLANK (USPTO)

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2000-20555

(P2000-20555A)

(43)公開日 平成12年1月21日(2000.1.21)

(51)Int.Cl.⁷

G 0 6 F 17/30

15/00

17/60

識別記号

3 1 0

F I

G 0 6 F 15/403

15/00

15/21

15/40

テマコード(参考)

3 4 0 A

3 1 0 A

Z

3 1 0 F

審査請求 未請求 請求項の数1 O.L 外国語出願 (全 98 頁)

(21)出願番号 特願平10-363765

(22)出願日 平成10年11月16日(1998.11.16)

(31)優先権主張番号 08-971,091

(32)優先日 平成9年11月14日(1997.11.14)

(33)優先権主張国 米国(US)

(71)出願人 597066212

アブテックス ソフトウェア インコーポ
レイテッドアメリカ合衆国 カリフォルニア州
92121 サン デイエゴ スクラントン

ロード 9605 スイート 240

(72)発明者 マイケル エ. ラザラス

アメリカ合衆国 92014 カリフォルニア,
デルマー カーメル パリー ロード ナ
ンバー 6 2222番地

(74)代理人 100089244

弁理士 遠山 勉 (外3名)

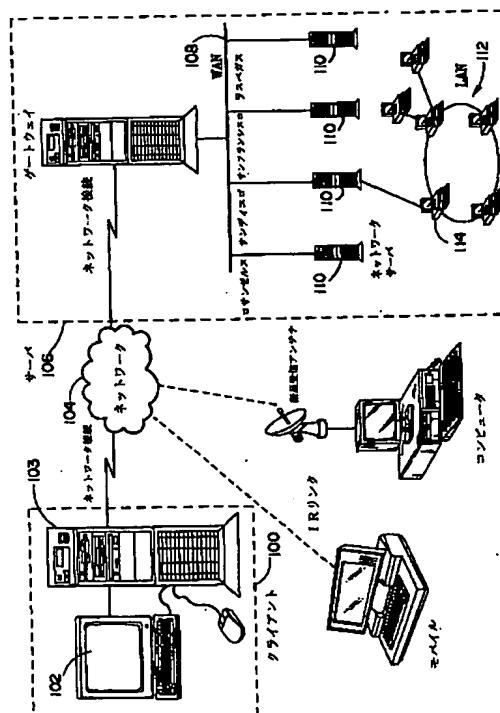
最終頁に続く

(54)【発明の名称】関連エンティティ及び関連情報へのリアルタイム最適化マッチング装置及び方法

(57)【要約】

【課題】追跡され、観察された、ユーザ一人一人に基づく行動および情報と行動の両方に適応性のあるベクトル空間表現に基づいて、広告、クーポン、製品および情報内容等の、個人的に対象付けられたエンティティを選択し、提示するシステムおよび方法を提供する。

【解決手段】本システムは、動作およびユーザ行動の観察が増加するに伴って向上する方法で、ユーザにエンティティをマッチングさせる。ユーザの行動およびエンティティ(宣伝、クーポン、製品)および情報(テキスト)はすべて均一なベクトル空間におけるコンテントベクトルとして表現される。本システムは、制約付きの自己編成学習技術を利用してシンボル(通常、構造化されていないテキスト中のワード)間の関係を学習するコンテンツベクトルと呼ばれる情報表現に基づく。ユーザおよびエンティティは、それぞれコンテンツベクトルとして表現される。



【特許請求の範囲】

【請求項1】 観察された行動を情報アイテムに関連づけるコンピュータ化されたシステムであって、観察された行動を行動ベクトルに変換可能なコンバータと、プロフィールベクトルを前記行動ベクトルで修正可能なプロフィールアダプタと、前記修正したプロフィールベクトルを、情報アイテムを示す複数のエンティティベクトルと比較して前記観察された行動と密に関連する少なくとも1つのエンティティベクトルを識別する比較可能な比較器と、を含むシステム。

【請求項2】 コンピュータ環境において広告を選択するシステムであって、電子広告のデータベースと、電子広告管理システムとを含み、該電子広告管理システムは、コンピュータ環境においてユーザコンピュータ装置の観察された行動を行動ベクトルに変換可能なコンバータと、前記ユーザを示すプロフィールベクトルを前記行動ベクトルで修正可能なプロフィールアダプタと、前記修正したプロフィールベクトルを、電子広告を示す複数のエンティティベクトルと比較して、前記観察された行動と密に関連する少なくとも1つのエンティティベクトルを識別する比較可能な比較器と、ユーザコンピュータ装置に通信する少なくとも1つの電子広告を選択するため、電子データベースに識別されたエンティティベクトルでアクセスするセレクタと、を含むシステム。

【請求項3】 エンティティベクトルを適応させるコンピュータ化されたシステムであって、ユーザの観察された行動を行動ベクトルに変換可能なコンバータと、前記行動ベクトルに基づいて、前記ユーザを示すプロフィールベクトルを修正可能なプロフィールアダプタと、前記プロフィールベクトルまたは前記行動ベクトルに基づいて、情報アイテムを示すエンティティベクトルを修正可能なエンティティアダプタとを含むシステム。

【請求項4】 コンピュータ環境においてプロフィールベクトルを生成するシステムであって、ユーザの観察された複数の行動を、関連する複数の行動ベクトルに変換可能なコンバータと、前記複数の行動ベクトルに基づいて、前記ユーザを示すプロフィールベクトルを繰り返し修正可能なプロフィールアダプタとを含むシステム。

【請求項5】 エンティティベクトルを適応させるコンピュータ化されたシステムであって、ユーザの観察された行動を行動ベクトルに変換可能なコンバータと、前記行動ベクトルに基づいて、情報アイテムを示すエンティティベクトルを修正可能なエンティテ

ィアダプタとを含むシステム。

【請求項6】 コンピュータ環境において広告を選択するシステムであって、電子広告のデータベースと、電子広告管理システムとを含み、該電子広告管理システムは、コンピュータ環境におけるユーザコンピュータ装置の観察された行動を行動ベクトルに変換可能なコンバータと、

前記行動ベクトルを、電子広告を示す複数のエンティティベクトルと比較して前記観察された行動と密に関連する少なくとも1つのエンティティベクトルを識別する比較可能な比較器とユーザコンピュータ装置に通信する少なくとも1つの電子広告を選択するため、電子データベースに識別されたエンティティベクトルでアクセスするセレクタと、を含むシステム。

【請求項7】 データ記憶装置を含むコンピュータ上で、観察された行動を情報アイテムと関連づける方法であって、

観察された行動を行動ベクトルに変換するステップと、プロフィールベクトルを前記行動ベクトルで修正するステップと、前記修正したプロフィールベクトルを、情報アイテムを示す複数のエンティティベクトルと比較して前記観察された行動と密に関連する少なくとも1つのエンティティベクトルを識別する比較ステップと、含む方法。

【請求項8】 コンピュータ環境において広告を選択する方法であって、

電子広告のデータベースを提供するステップと、コンピュータ環境においてユーザコンピュータ装置の観察された行動を行動ベクトルに変換するステップと、前記ユーザのプロフィールベクトルを前記行動ベクトルで修正するステップと、前記修正したプロフィールベクトルを、電子広告を示す複数のエンティティベクトルと比較して前記観察された行動と密に関連する少なくとも1つのエンティティベクトルを識別する比較ステップと、前記識別されたエンティティベクトルで電子データベースにアクセスするステップと、前記ユーザコンピュータ装置と通信するため少なくとも1つの電子広告を選択するステップと、を含む方法。

【請求項9】 データ記憶装置を含むコンピュータ上で、エンティティベクトルを適応させる方法であって、ユーザの観察された行動を行動ベクトルに変換するステップと、

前記行動ベクトルに基づいて、前記ユーザを示すプロフィールベクトルを修正するステップと、前記プロフィールベクトルまたは前記行動ベクトルに基づいて、情報アイテムを示すエンティティベクトルを修正するステップとを含む方法。

【請求項10】 コンピュータ環境において、データ記憶装置を含むコンピュータ上でプロフィールベクトルを

生成する方法であって、
ユーザの観察された複数の行動を、関連する複数の行動ベクトルに変換するステップと、
前記複数の行動ベクトルに基づいて、前記ユーザを示すプロフィールベクトルを繰り返し修正するステップとを含む方法。

【請求項11】 データ記憶装置を含むコンピュータ上でエンティティベクトルを適応する方法であって、
ユーザの観察された行動を行動ベクトルに変換するステップと、
前記行動ベクトルに基づいて、情報アイテムを示すエンティティベクトルを修正するステップと、を含む方法。

【請求項12】 データ記憶装置を含むコンピュータにおいて広告を選択する方法であって、
電子広告のデータベースを提供するステップと、
コンピュータにおけるユーザコンピュータ装置の観察された行動を行動ベクトルに変換するステップと、
前記行動ベクトルを、電子広告を示す複数のエンティティベクトルと比較して前記観察された行動と密に関連する少なくとも1つのエンティティベクトルを識別する比較ステップと、
前記識別したエンティティベクトルで電子データベースにアクセスするステップと、
前記ユーザコンピュータ装置と通信するため、少なくとも一つの電子広告を選択するするステップと、を含む方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明はエンティティおよび情報を行動と関連づけることに関する。より詳細には、本発明は観察された行動に基づいて広告、クーポン、製品または情報内容をユーザに的をしぼり、かつ配信するシステムおよびプロセスに関する。

【0002】

【従来の技術】 ワールドワイドウェブ（ウェブ）およびインターネットサービスの広範囲にわたる可用性は、結果として独自な形式の広告の機会をもたらしている。テレビやラジオ等の従来の「ブロードキャスト」メディアとは異なり、ウェブはユーザが受信情報をより高度に制

御することが可能な「ナロー・キャスト」メディアである。ユーザが情報の検索を制御できるため、従来のブロードキャストモデルで利用される広告技術はより効果的ではなくなり、また「ショットガン」効果の結果、潜在的な顧客を遠ざけることもある。ユーザ一人一人を基準としたベースで選択的に広告をユーザに向けることの可能性は、有意義な顧客の対象付けを行うことが困難であるため、実現されていなかった。ウェブ広告選択エンジンの現世代は、顧客を対象付けるのに部分的に成功している多数の技術を利用している。典型的に、これらの技術の有効性は、視聴者の応答率によって測定される。

「クリックスルー」とも呼ばれる視聴者応答は、ユーザに提示されたウェブページに含まれる「見出し広告」をクリックするユーザ数をカウントすることによって評価される。見出し広告をクリックすると、通常ユーザは、製品またはサービスについてのさらなる情報が提供される広告主のウェブサイトに運ばれる。現在のインターネット広告環境においては、クリックスルーが広告技術の有効性の最善の測定方法である。結果として、広告の価値は、その広告の有効性に直接関連する。したがって、クリックスルーの最大化が、実際的かつ経済的の両方の理由により、大部分のウェブサイトにとって最も重要である。

【0003】 いくつかの現在の見出し広告選択技術を、以下の表にリストする。これらの技術は2つのクラスに分類される。表1に示す広告選択技術の第一のクラスは、不变の以前の情報のみに基づく単純化した第一世代の技術を用いる。表2に示す広告選択技術の第二のクラスは、ユーザ照会等のユーザ行動のある測定を考慮したり、あるいは事前に明示されたユーザの志向を利用する、より技巧をこらした第二世代の広告選択システムにおいて使用される技術のいくつかを利用している。しかしながら、これらより技巧をこらした目的は、多くの場合、問題を複雑にするだけである。表1および表2において、各方法に特定な不都合点が提示されているが、この表はすべてを包括するものではなく、すべての不都合点が示されているわけではない。

【0004】

【表1】 第一世代広告対象付け技術

方法	不都合な点
ユーザのブラウザのドメイン名	<ul style="list-style-type: none"> ・有意なユーザの洞察を提供しない、ユーザの行動がわからない ・ユーザの行動が使用ドメインに関連することを示す ・クリックスルーに乏しい
ブラウザタイプ	<ul style="list-style-type: none"> ・ユーザの行動が使用ブラウザタイプ/バージョンに関連することを示す ・クリックスルーに乏しい
システムタイプ	<ul style="list-style-type: none"> ・ユーザの行動が使用システムタイプに関連することを示す ・クリックスルーに乏しい
サービスプロバイダ	<ul style="list-style-type: none"> ・ユーザの行動が使用ドメインに関連することを示す ・クリックスルーが乏しい
地理/位置	<ul style="list-style-type: none"> ・場所に対するIPアドレスの最新の知識ベースが必要 ・クリックスルーに乏しい ・有意なユーザの洞察を提供しない
サイトSICコード	<ul style="list-style-type: none"> ・IPアドレスに対するSICコードの最新の知識ベースが必要 ・いくらか改善されたユーザのクリックスルーを提供する ・有意なユーザの洞察を提供しない
企業の規模	<ul style="list-style-type: none"> ・IPアドレスに対するSICコードの最新の知識ベースが必要 ・有意なユーザの洞察を提供しない ・ユーザの行動が企業の大きさに関連することを示す
知識ベース技術	<ul style="list-style-type: none"> ・知識ベースの維持が必要 ・維持費がかかる ・観察されたユーザの行動を組み込むことができない

【0005】

【表2】 第二世代広告対象付け技術

方法	不都合な点
ユーザ登録情報	<ul style="list-style-type: none"> ・情報を開示するユーザの意志が必要 ・効果的な動作には製品についての多くの判断が必要 ・相当な既存の判断データベースが必要
探索キーワード	<ul style="list-style-type: none"> ・概念的に近いがキーワードを含まない広告を見落とす ・利用可能な在庫目録を制限する ・注意深いキーワードの選択が必要
手動で広告をサイトまたはページに対象付ける	<ul style="list-style-type: none"> ・非常に労働集約的、正しく見積もららない ・人間は多数の広告を手動で対象付けることが得意ではない
手動で解析したサイトまたはページに対象付ける	<ul style="list-style-type: none"> ・非常に労働集約的、正しく見積もららない ・人間は多数のサイトを手動で解析することが得意ではない
無作為提示	<ul style="list-style-type: none"> ・応答率が低い ・ユーザを遠ざける
日付／時間	<ul style="list-style-type: none"> ・応答率が低い
手動で解析したサイト	<ul style="list-style-type: none"> ・非常に労働集約的 ・高コスト

【0006】現在のウェブ環境では、ユーザは関連するまたは関心のある情報を見つけるために、探索サービスを使用する。これらの探索サービスは、実行する探索によって限定されるユーザ行動、ユーザが閲覧するウェブページおよび訪れるディレクトリ階層の位置の識別について可能性のありそうな絞り込みを行う。しかし、既存の見出し広告選択技術は、広告を選択する際にこの行動を解析しない。これらの探索サービスによって、ユーザ特定の広告を提示する機会が与えられる。

【0007】表1および表2に示す従来技術は、顧客を「対象付け」るために、手動でコード化した関連性およびリストを使用する。ある技術は、広告対象付けの一部として現在のユーザ照会のみを使用する。表1および表2に示す各方法は、5つの一般的な広告選択技術のカテゴリに関連する。これらのカテゴリには、以下が含まれる。

(i) 法則に基づく類別

(ii) キーワードに基づく広告選択

(iii) 個々のウェブページまたはサイトへの広告割り当て

(iv) ページの階層組織におけるブランチへの広告割り当て

(v) 共同フィルタリング

各カテゴリの説明を以下に述べる。

【0008】(法則に基づく広告選択) 法則に基づく広告選択は、適切な広告を選択するために、利用可能な情報および法則を使用する。この技術は、多数の法則が手動でコード化され、法則開発者が問題の領域に深い理解を有する場合に効果的である。しかし、このシステムの

一つの问题是、法則を書くまたは維持するために、人間の知的努力が必要なことである。法則基礎の開発は、非常に高価であり、時間のかかるものである。

【0009】さらに、広告選択のための法則は、夫々個の値を有する利用可能な変数に限定され、「不確実な」決定境界を提供する。これらの決定境界は通常2進法であり、取り扱いを誤る場合が多い。

【0010】法則に基づく広告選択には、対象付けられた動作領域についての広範囲にわたる知識が必要である。コンピュータ支援されたツールを用いても、法則を開発しシステムを管理するために知識エンジニアが必要である。さらに、人間は、法則をコード化することに不得意であることを何度も実証してきた。この観察は、多数の変数が、モデル化される問題の範囲に含まれる場合に、特に当てはまる。

【0011】(キーワードに基づく広告選択) キーワードに基づく広告選択のシステムでは、広告はユーザが提供した一つ以上のワードを基にして選択される。観察されたユーザ行動（典型的にはユーザが照会を発する）が周知のキーワードを含む場合、手動でキーワードに関連する広告の一つが選択されて表示される。この技術では、選択されたキーワードに対しての応答率はよい。しかしながら、このアプローチの主な欠点は、システム管理者が各広告に関連するキーワードを手動で選択しなければならないことである。この技術は、知的労力および広告特定の領域の深い知識に基づき、時間がかかり、かつ間違いやすい。さらに、サイトでのキーワードの「在庫目録」は、すぐに売り切れてしまう。最後に、キーワードに基づく広告選択技術では、広告選択プロセスは以

前のユーザの行動を考慮しない。広告セレクタは、現在の探索照会に基づく現在の在庫目録中の、人間が選択したキーワードのセットのみを使用する。ユーザが与えたキーワードの類義語は、類義語辞典または類義語リストなしでは、自動的には対象付けられない。

【0012】（個々のページへの広告割り当て）広告は時には個々のウェブページに手動で対象付けられる。この方法には、広告をウェブページにマッチさせるのに人間の知的努力が必要である。しかし、このような努力は、通常数千のウェブページを含む大規模のサイトにおいては無理がある。

【0013】（ページの階層を有するプランチへの広告割り当て）広告は時には手動でページ存在論と呼ばれるページの階層またはカテゴリに対象付けられる。この技術は、階層のプランチへの広告の割り当てが、サイトのコンテンツおよび想定されるピュアの理解に関して良い感触を持つものである場合、有効に働く。しかし、広告に適切なプランチまたはカテゴリを選択するために、人間の知的努力が必要である。

【0014】（ウェブサイトへの広告割り当て）広告は時として手動でウェブサイトのすべてのページに対象付けられる。しかしながら、他のシステムの多くに見られたように、ウェブサイトに最適な広告を選択するために人間の知的努力が必要である。この方法は、広告主が自動的に広告をサイト内の最適なページに対象付けできないという点において柔軟性に欠ける。通常、広告主はサイト中の各ページに異なる広告を表示したいと望む。ウェブサイトへの広告割り当てに伴う別の問題は、ユーザのための広告が、ユーザ特定の情報を利用できないという点において情報依存症ではないことがわかる。

【0015】（共同フィルタリング技術）共同フィルタリング（CF）技術は、選択的に特定の広告をウェブユーザに対象付けるという課題のために提案された。CFアプローチでは、多数のユーザが形式的にシステムに登録し、ユーザが受信する広告、クーポンおよび情報内容の品質について選択の判断を行う必要がある。特定の限定された環境では、この技術は、小規模のテストにおいてある程度有効なマッチングをもたらす。しかしながら、インターネットに見られるような大きなタスクについて、そのアルゴリズムは計算的に処理しにくくなると共に非実用的になる。さらに、CF技術は、エンティティのユニバースが時間と共に発展するよりもむしろ全く変化しない場合に最もよく機能する。インターネット市場において、全く変化しないということは例外であり、一般ではない。さらに、CFアプローチの有用性は、選択判断を行ったユーザの数に直接関係する。個人の時間の制限とプライバシーの考慮のために、通常、ユーザはこれらの判断を行うのに必要な相当量の努力を費やすことに気が進まない。従って、データ獲得は広告主にとって主要な問題となる。

【0016】CF解決法は通常、問題空間の次元が考慮するエンティティの数に等しい、非常に高次元の空間ベクトルのシステムを実行するために、知識エンジニアが必要である。しかしながら、この解法は、計算的に処理しにくいことが判明している。CF技術の別の問題は、相当数の統計的に複雑なステップが必要である直交関係を広告選択ソフトウェアが検査することである。広告選択ソフトウェアはベクトル空間において近似しておりかつ近似するものの間で座標が共通していないものを見つけなければならない。

【0017】既存の広告選択システムは、特にクーポンおよびプリントメディア広告において、インターネットおよび電子商取引の対話的性質に注意を向けていない。広告主は特定の関心を有するユーザを識別し、経時にわたってそれらの関心を追跡し、非常に選択的な方法で、ユーザが関心を持つ情報、広告、クーポンおよび製品の提供を流布する必要がある。さらに、広告主はユーザの関心および行動をリアルタイムで追跡する必要がある。

【0018】

【発明が解決しようとする課題】したがって、広告主は、対象付けられている広告、クーポン、製品および情報内容について、ユーザ行動を感知するシステムを必要としている。このシステムによって、リアルタイムで、および同様の行動特徴を有するユーザグループと同様に個人ユーザそれぞれへ細かく対象付けた販売広告が可能となる。

【0019】

【課題を解決するための手段】本発明は、観察された行動の追跡に基づいて、ユーザに広告、クーポン、製品および情報内容等の対象付けられたエンティティを配信するシステムおよび方法を含む。システムモデルはエンティティ、情報および行動についての適応可能なベクトル空間表現である。システムは動作およびユーザ行動の観察の増加と共に改善されるように、ユーザをエンティティにマッチさせる。

【0020】本発明はエンティティまたはコンテンツベクトルと呼ばれる情報的な表現を使用する。コンテンツベクトルは、前後関係の類似性をコード化する高次元のリアルなベクトルの表現である。システムは、典型的に構造化されていないテキストにおけるワードであるシンボルの間の関係を学習するため、制約的な自己組織化学習アルゴリズムを利用する。この学習アルゴリズムは、興味の領域についてのトレーニング例の集合からのワードまたはシンボルの間の関係を発見し、各シンボルに固有のコンテンツベクトルを割り当てる。結果として得られる表現は、類似する前後関係で現れるシンボルが高次元コンテンツベクトル空間において同様の方向を指すコンテンツベクトルを所有する。

【0021】ウェブページ、見出し広告、クーポンおよび他のあらゆるテキストまたはシンボルの情報は、ペー

ジ、あるいは製品またはサービスの説明でのワード（シンボル）のコンテンツベクトルの正規化加重ベクトル和を形成することによって、サマリーコンテンツベクトルとして表される。これらのサマリーコンテンツベクトルは類似する情報を説明するページが同様の方向を指すサマリーコンテンツベクトルを有する特性を有する。

【0022】各ユーザの動作は、プロフィールベクトルで表わされる。典型的なユーザの動作には、ページの要求または読み込みの照会の発行、インターネットサイトの訪問、広告に対しての応答、およびクーポンの引き替え、製品の購入が含まれるが、これらに限定されない。ユーザのプロフィールベクトルは、関連する動作のコンテンツベクトル表現を使用することによって、観察された行動を追跡するよう構成される。たとえば、ユーザが情報ページを要求した場合、ユーザのプロフィールベクトルはその要求されたページのエンティティベクトルに基づいて調整される。

【0023】ユーザプロフィールベクトルは、ユーザに提示するのに最もよいエンティティ（たとえば、広告、クーポン、製品または情報）を選択するメカニズムとして利用される。ユーザプロフィールベクトルは、各アクティブエンティティベクトルと比較され、最も近いエンティティコンテンツベクトルが選択されてユーザに表示される。

【0024】また、本発明は、エンティティベクトルの最適な適応配置を行う。ユーザに見出し広告が提示され、ユーザがその上でマウスをクリックしてさらなる情報を要求した場合、その広告のエンティティベクトルはユーザのプロフィールベクトルを使用して調整される。さらに、ユーザプロフィールベクトルは、その広告コンテンツベクトルを使用して調整される。この相互適応によって、ユーザに対しての広告の最適な配置が可能となる。

【0025】本システムにおいて、ユーザプロフィールベクトルは、ユーザの本システムの使用によって発展する。クラスタ化および分類技術が、集中した販売キャンペーンに利用される行動パターンを発見するため、適応されたプロフィールベクトルに適用される。

【0026】同様に、広告およびクーポンエンティティベクトルは、ユーザ行動の結果、発展する。これらの発展したベクトルは、広告をクリックしたまたはクーポンを引き替えたユーザの特徴の要約をコード化する。これらの特徴の要約は、製品配置を修正し、その製品またはサービスに興味を持つユーザの特徴を発見するために使われる。

【0027】システムは、完全に自動化され、動作についての法則を必要とせず、自己組織ニューラル・ネットワーク技術に基づく。さらに、システムは暗黙のユーザ行動を利用して動作し、正確な操作のためのユーザの明白な判断は必要ではない。

【0028】

【発明の実施の形態】本発明のある具体的な実施形態の詳細な説明を以下に示す。しかしながら、本発明は請求の範囲により定義されかつカバーされる多数の異なる方法で具体化することができる。本説明では図面を参照し、図面全体を通して同様の部分は同様の番号で示されている。

【0029】図1は、本発明に関連するコンピュータ環境を示すブロック図である。クライアントコンピュータ100は、モニタ102および処理装置103を有する。処理装置103にはデータ記憶装置が含まれる。

【0030】クライアントコンピュータ100は、ネットワーク104を介して他のコンピュータ装置に伝えられる情報を記憶する。ネットワーク104は、たとえば、インターネット、イントラネット、ローカル・エリア・ネットワーク（LAN）またはワイド・エリア・ネットワーク（WAN）等のネットワークを含む、電気的に接続されたあらゆるタイプのコンピュータ群を含むことができる。さらに、ネットワークへの接続方法は、たとえば遠隔モ뎀、イーサネット（IEEE 802.3）、トーカンリング（IEEE 802.5）、光ファイバ分散データリンク・インターフェース（FDDI）または非同期転送モード（ATM）である。コンピュータ装置はデスクトップ、サーバ、ポータブル、ハンドヘルド、セットトップ、あるいは他のいかなるタイプの構成でもよいことに留意する。

【0031】遠隔ユーザがネットワーク104から情報を要求すると、クライアントコンピュータ100が要求された情報とともに広告を表示する。クライアントコンピュータ100からの要求に基づき、サーバ106はユーザの観察された行動に基づいてユーザに表示するのに適切な広告を選択することができる。

【0032】一実施形態において、サーバ106はWAN108に接続されているゲートウェイを含む。WAN108は、複数のネットワークサーバ110を有する。ネットワークサーバ110のうちの一つは、複数のコンピュータ114を含むLAN112に接続されている。広告を選択するシステムソフトウェアは、ネットワークサーバ110の一つまたはネットワーク104中の他のコンピュータに配置されてもよい。本発明の一実施形態において、広告選択ソフトウェアは、一部は複数のネットワークサーバ110で実行される。本発明の別の実施形態においては、広告選択ソフトウェアはLAN112の複数のコンピュータ114で実行される。本発明のさらに別の実施形態においては、広告選択ソフトウェアはクライアントコンピュータ100に常駐する。通信経路が広告選択ソフトウェアと広告の要求者との間に存在する限り、本発明の広告選択システムはいかなるコンピュータ装置上でもホストされうることを理解することが重要である。

【0033】図2は、本発明の処理およびデータ構成要素を示す機能的なブロック図である。以下の説明は、インターネット上で広告、特に見出し広告を配信するシステムの能力に焦点をあてる。しかしながら、本発明で利用する技術は性質が一般的であるため、あらゆるタイプのエンティティまたは情報に適用することができる。さらに、ネットワーク環境は前述のようなインターネットに限定されない。

【0034】本発明は、情報項目またはエンティティベクトル間の関係を識別するために、情報のベクトルのモデル化を用いる。エンティティベクトルを作成および使用するプロセスは、カイド他 (Caid et al) に付与された「SYSTEM AND METHOD OF CONTEXT VECTOR GENERATION AND RETRIEVAL」と称する、1995年11月21日に出願された米国特許第5, 619, 709号に開示されている。

【0035】このエンティティベクトル生成スキームは、コンテンツベクトルを生成するよう設計されている。たとえばワードへの応用では、このスキームは一語につき一つのベクトルを生成し、これは量および幾何学的な方法においてワードまたは文書間の意味や内容の相対的な近似を表す。次いでシステムは、エンティティにおけるワードおよびそれらの関連するワードコンテンツベクトルに基づいてエンティティベクトルを作成する。したがって、類似する意味を持つ情報アイテムは近似して並んだベクトルを有する一方で、類似しない意味を有する情報アイテムはおおよそ直交するベクトルを有する。

【0036】以下の説明において、「コンテンツベクトル」と「エンティティベクトル」という言葉は同義で用いられる。両用語は、たとえばある内容情報、広告、クーポン、勧誘または製品に関連するベクトルをいう。

【0037】以下の説明は、(1) システム構成要素概説、(2) ワードコンテンツベクトル学習モジュール、(3) セッションコンパイラモジュール、(4) リアルタイム広告サーバモジュール、(5) プロフィールベクトル更新サーバモジュール、(6) エンティティベクトル更新サーバモジュール、(7) プロフィールおよびエンティティ適応アルゴリズム、(8) エンティティインベントリ加重計算(コストモデル)、および(9) 関係発見および解析処理モジュール、に分類される。

【0038】(システム構成要素概説) 図2の下部を参照して説明を開始すると、ワードコンテンツベクトル学習モジュール200は、領域関連トレーニングコーパス202、ストップリストファイル204、およびフレーズリストファイル206からワードコンテンツベクトルの組を作成する。ストップリストファイル204は、ワードコンテンツベクトル学習モジュール200が無視するワード、ワードの断片およびフレーズのリストを含む。フレーズリストファイル206は、ワードコンテン

トベクトル学習モジュール200と一緒にグループ化するワードのリストである。トレーニングコーパス202は、ワードベクトルをトレインするのに用いられる一組の情報オブジェクトである。トレーニングコーパス202、ストップリストファイル204およびフレーズリスト206の機能を、さらに以下で述べる。

【0039】ワードコンテンツ学習プロセスが完了すると、トレーニングコーパス202にはあり、ストップリストファイル204にはないワードのそれぞれにつき1つのワードコンテンツベクトルがワードベクトルデータベース208に存在する。これらのワードコンテンツベクトルは、同様の文脈で用いられるワードが高次元コンテンツベクトル空間において同様の方向を指すコンテンツベクトルを有する特性を有する。ワードコンテンツベクトルは、本発明によって動作中に行われる広告選択動作のベースとして機能する。ワードコンテンツベクトル学習プロセスは、初期化プロセスの一部として行われる。しかしながら、これらのベクトルは所望ならば更新することができる。通常、ワードコンテンツベクトルは、英語におけるワードの意味は時を経て徐々に進化するため、本発明の正常動作中は変化しない。

【0040】セッションコンパイラモジュール210は、セッション記述データベース212から構造化されたテキストの形でセッション制御情報の組を取り、そのテキストを、セッション制御ファイル215のリアルタイム広告サーバ(RTAS)モジュール214の制御コマンドに変換する。コンパイルプロセス中に処理される典型的な情報には、アクティブな広告の数、各広告についての許容可能な印刷通し数、スタートおよびストップの回数、および各広告を掲載する日数が含まれる。セッションコンパイラモジュール212は、その機能が必要でない場合に、システムから省くことができる。

【0041】次いで、本発明は各広告の初期エンティティベクトルまたはシステム中のエンティティを計算する。ほとんどの場合、このエンティティベクトルは、エンティティ記述データベース216中に位置するエンティティのテキスト記述に基づいている。たとえば、エンティティが車の広告である場合、その広告中のワードがエンティティベクトルを決定するために用いられる。エンティティベクトルは、広告にあるワードコンテンツベクトルの正規化加重和として計算される。あらゆるアイテムのテキスト記述は、そのエンティティベクトルの初期状態として使用することができる。エンティティベクトルはシステム動作中に進化しうるため、エンティティベクトルの初期状態は空白であることもあり、またエンティティのテキスト記述はシステム動作に必要ではない。しかし、正常動作において、この進化は、システムの使用率が低い場合に相当な時間を必要とし、その結果、好ましい実施形態は初期状態として適当なテキスト記述で始まる。

【0042】本発明では、すべての広告が正確なマッチではなくベクトルの比較を基にして選ばれるため、従来のシステムよりも多くの広告を関心のあるテーマ領域に販売できるので利用可能な在庫を増加させる。たとえば、本発明のプロバイダは、ただ選択したワードのセットではなく、広告中のワードに近いベクトルハイパスベース領域に基づいて広告枠を売ることができる。

【0043】また、エンティティベクトルアプローチによって、管理者が特別な技術または領域の知識なしでも広告定義を効果的にセットアップすることができ、このため領域の専門家を必要とせずにシステム全体の効率を向上することができる。本発明はただサイト全体ではなく、広告をサイト中の最適なページに自動的に対象付ける。この特徴の結果、異なる広告が1つのサイトの同じページに属することもある。さらに、本発明は階層またはカテゴリ内で、特定の広告にもっとも関連するページを自動的に選択する。あるいは、システムを、存在するであろうし、かつ観察されたユーザ行動へのマッチングにのみ基づいた存在論に関係なく、最適な広告を供給するよう構成することもできる。

【0044】本発明では、ユーザに提供される広告は、ページの内容、および現在のユーザ動作と過去のユーザプロフィールの両方に依存している点において、内容依存といえる。エンティティベクトルアプローチは、コンパクトで適応性のある解決法を提供する。共同フィルタリングアプローチと比較して、エンティティベクトル表現には、必要とされるトレーニングのためのデータは非常に少量である。さらに、共同フィルタリングアプローチと比較して、本システムの多次元エンティティベクトル表現により、優れた一般化能力がもたらされる。

【0045】再び図2を参照して、リアルタイム広告サーバ(RTAS)モジュール214はユーザ行動を観察するエンジンであり、その行動に基づいて図の上部に位置するユーザコンピュータ219に提示するのに適切な広告を選択する。RTASモジュール214は、適切な広告を選択するのに、現在のユーザ行動、ユーザのプロフィールベクトル、およびエンティティベクトルのアクティブなセットを用いる。RTASモジュール214は、ユーザに表示するのに最適な広告を決定するために、内積等のベクトル近似計算を用いる。しかしながら、他の距離測定基準も使用することができる。

【0046】RTASモジュール214が、ユーザがクライアントコンピュータ219上で照会の発行(217)、ページの要求(218)、製品の購入または広告のクリック(222)等の「適応イベント」を行ったことを感知した場合、RTASモジュール214は更新メッセージをプロフィール更新サーバ(PVUS)モジュール224またはエンティティ更新サーバ(EVUS)モジュール228に対して送信する。RTASモジュール214は、スタティックコンテンツベクトルデータベ

ース234および高速ファイル232を利用して、エンティティまたは広告ベクトルの作動値を保持する。適切なエンティティを選択した際、RTASモジュール214は選択した広告をクライアントコンピュータ219に送信する。

【0047】PVUSモジュール224は、RTASモジュール214からの要求に応答してユーザのプロフィールベクトルを適応させる。ユーザそれぞれのプロフィールベクトルはプロフィールベクトルデータベース226に格納される。この適応プロセスは、ユーザプロフィールベクトルを適応イベントと関連するエンティティベクトルの方向に移動する。たとえば、ユーザが広告をクリックした場合、そのユーザのプロフィールベクトルはその広告のエンティティベクトルの方向に少し移動する。

【0048】EVUSモジュール228は、PVUSモジュール224とほぼ同一の動作を行うが、広告のエンティティベクトル上で動作する。各エンティティベクトルは、エンティティベクトルデータベース230に格納されている。たとえば、ユーザが広告をクリックした場合、そのユーザが選択した広告のエンティティベクトルがそのユーザのプロフィールベクトルの方向に少し移動する。

【0049】関係発見および解析(RDA)モジュール236は、プロフィールベクトルデータベース226とエンティティベクトルデータベース230の両方の内容に関して処理動作を行って、有用なパターンおよび傾向を識別する。RDAモジュール236は、有用な行動パターンを識別するため、エンティティベクトルとプロフィールベクトル両方のクラスタ化を利用する。

【0050】エンティティベクトルの適応はシステムの動作に必要とされないことに留意すべきである。まさに、適応が可能となった場合、それによって製品のエンティティベクトルが広告を受け取ったユーザの方に「ドリフト」することが可能となる。ある用途、特に情報内容配信システムでは、このドリフトは望ましくない。明らかに、ニュースの情報内容は決定的であり、誰がこのニュースを読むかによって変わるべきではない。このように、本発明はエンティティベクトルの適応を不能にするための設備を有する。しかし、インターネット広告等の多くの用途において、ベクトルのドリフトは、システムがパターンおよび傾向を発見するようなメカニズムである。

【0051】本明細書に記載する好ましい実施形態において、ソフトウェアプログラムはC、C++およびTCLプログラム言語で書かれており、周知のUNIX作動システムの下で実行される。C、C++、TCLは、業界標準のプログラム言語であり、実行可能コードを作成するために、多くの市販されているコンパイラを用いることができる。しかし、本発明では、従来の様々なハー

ドウェアおよびソフトウェアプラットフォームで実行することができる。本明細書に記載するソフトウェアシステムは、異なるハードウェアプラットフォームで実行している異なるサブシステムにわたって分散することもできる。構成要素は、伝送制御プロトコル／インターネットプロトコル(TCP/IP)およびRCP等のネットワークプロトコルを使用して伝送されるメッセージを用いて通信することができる。これらのモジュールおよびファイルをホストする具体的なプラットフォームは、ホストコンピュータ環境に十分な処理能力がある場合、本発明にとっては重要ではない。さらに、処理およびデータ構成要素のそれぞれは、接続のパスが十分確かな通信プロトコルを介して提供される限り、別個のプラットフォーム上でホストされてもよい。

【0052】本発明は、以下の6つの原則的な処理モジュールを含む。すなわち、ワードコンテンツベクトル学習モジュール200、セッションコンパイラモジュール210、リアルタイム広告サーバモジュール214、エンティティベクトル更新サーバモジュール228、プロフィールベクトル更新サーバモジュール224、および関係発見および解析モジュール236である。以下に、上記の構成要素をさらに詳細に説明する。

【0053】(ワードコンテンツベクトル学習モジュール)ワードコンテンツベクトル学習モジュール200(図2に示す)は、アプリケーションプログラムインターフェースとして実施される。アプリケーションプログラムインターフェースは、コンピュータプログラムライブラリの一部であってもよい。広告者はこのライブラリが提供する手順を利用して、十分な処理能力を有するプラットフォーム上で独立型で実行可能なコンピュータプログラムを作成する。このプログラムは、一実施形態では前記カイドの特許で更に記載されている、コンテンツベクトル学習プロセスを行う。さらに、このプログラムは、エンティティベクトル学習の出力をデータ構造中に書き込み、このデータ構造は、次いで、実行時間の効率について最適化されたファイルを生成するため、セッションコンパイラモジュール210によって使用される。

【0054】構造化されていないテキストまたはシンボルを包括する多数の文書を含むトレーニングコーパス202が選択され、コンテンツベクトル学習アルゴリズムを介して処理される。次いで、コンテンツベクトル学習プロセスを介して生成された、結果のオブジェクトは、他のシステム構成要素による使用に適応される。

【0055】図3は、図2に示すワードコンテンツ学習モジュール200の前処理モジュール300の入力および出力を例示するブロック図である。前処理の一部として、ワードコンテンツベクトル学習は、以下の機能を行う。

【0056】1. 入力の形成

図3を参照して、ワードコンテンツベクトル学習モジュ

ール200に必要な入力ファイルがアセンブルされる。3つの入力ファイルは、トレーニングコーパス202、ストップリストファイル204およびフレーズリストファイル206である。

【0057】対象となるシンボルが一連のワード301である場合、トレーニングコーパス202は、関心のある領域をカバーする構造化されていない自由形式のテキスト文書のセットを含む。たとえば、関心のある領域がハンティングや釣り等のアウトドアスポーツである場合、適切なトレーニングコーパスは、「Field and Stream」という雑誌からの3年間の記事および地方紙のアウトドアセクションである場合もある。このテキストは一般に、植字やフォーマット情報が取り除かれた「フラット」なASCIIフォーマットである。

【0058】ストップリストファイル204(図2および図3に示す)は、人間の知的な努力と従来の統計的な技術の組み合わせによって開発され、ワードコンテンツベクトル学習モジュール200が無視するワード、ワードの断片、およびフレーズのリストを含む。コンテンツベクトルワード学習に関して、従来の統計的な情報検索技術のように、多くのワードは意味のないものである。たとえば、「および(AND)」、「ない(NOT)」「しかし(BUT)」および「その(THE)」はテキストのセットのテーマの内容を判定するという点では、ほとんど意味を伝えない。ゆえに、これらの用語は無視されるべきである。ストップリストファイル204は、一行につき一つのエントリを有する、多数の列挙された無視すべきワードのリストを有する、フラットASCIIファイルである。従来の用途のためのこのファイルの典型的な長さは、トレーニングコーパス202における統計的に最も頻繁に発生するワードの約250～350ワードである。

【0059】フレーズリストファイル206(図2および図3に示す)は、単一シンボルとして扱われるべきワードのシケンスのリストを含む。このファイルを使用して、ワードコンテンツベクトル学習モジュール200は、特定のフレーズを単一シンボルとして取り扱う。たとえば、「San Diego」は単一シンボルとして考慮されるべきである。フレーズリスト206の内容は、知的な努力、統計的な技術、言語処理またはこれらの組み合わせによって決定される。フレーズリストファイル206は、ワードコンテンツベクトル学習モジュール200の正確な動作には必要ではないが、有意なフレーズリストを使用することで、システムの性能が向上する。フレーズリスト206は、ファイル中の一行につき一つフレーズあるフラットASCIIフォーマットである。

【0060】2. ワードコンテンツ学習の実行
トレーニングコーパス202、ストップリストファイル204およびフレーズリストファイル206が一旦アセンブルされると、ワードコンテンツベクトル学習プロセ

スが行われる。前記カイド特許はこの方法についての完全な開示を提供するが、便宜上重要なステップの概要をここで繰り返す。

【0061】図3を参照して、前処理が、ワードコンテンツ学習モジュール200の一部である前処理モジュール300によって行われる。前処理モジュール300は、トレーニングコーパス202を読み、現行のワードがフレーズリスト206またはストップリストファイル204にあるかどうかを判定し、そのワードの語幹化を行い、そして結果として得た有効な語幹302を中間語幹ファイル304（図3に示す）に書き込む。語幹化プロセスは、トレーニングコーパス202中のワードのそれぞれから共通の接尾辞を取り除く。

【0062】図4は、語幹ハッシュテーブル400を作成するプロセスの入力および出力を例示するブロック図である。語幹ハッシュテーブル400は、語幹ハッシュテーブル作成モジュール402によって作成される。語幹ハッシュテーブル作成モジュール402は、ワードコンテンツベクトル学習モジュール200の副構成要素である。中間ファイル304内のすべての語幹302について、データアイテム404が語幹ハッシュテーブル400において作成される。ハッシュテーブルはまた、各語幹についての、語幹テキストストリングからなる固有のデータ要素へのポインタ、語幹統計、語幹コンテンツベクトル、および学習プロセスの一部として語幹コンテンツベクトルに適用される変更ベクトルを蓄積するために一時的な記憶装置として使用される訂正コンテンツベクトルから構成される。

【0063】図5は、語幹ベクトルの初期状態を作成する、乱数生成プロセスを例示するブロック図である。語幹ハッシュテーブル400中の各語幹データアイテムについての初期状態は、均一な擬似乱数生成（UPNG）モジュール500によって作成される。UPNGモジュール500は、初期無作為コンテンツベクトルを生成し、このベクトルを語幹コンテンツベクトルに格納する。UPNGモジュール500は、均一分布乱数502を作成する長期擬似乱数生成装置を使用することによって、語幹ベクトルの初期状態を生成する。繰り返し期間が十分長い（100,000,000サイクル以上）と仮定して、任意の乱数生成装置を使用することができる。均一分布乱数502を用いて、変換モジュール504は、ゼロ平均および単位分散の特性を有するガウスのランダム変量502を作成する。インストレーションモジュール508は、現在の語幹コンテンツベクトルの第一の要素に第一のガウスのランダム変量を割り当てる。次いで、第二のランダム変量は語幹コンテンツベクトルの第二の要素に割り当られる。この固有のガウス分布乱数を割り当てるプロセスは、語幹ハッシュテーブル400中のすべての語幹のすべての語幹コンテンツベクトルが埋まるまで繰り返される。

【0064】図6は、図2に示すコンテンツベクトル学習モジュールの語幹頻度統計計算プロセスを例示するブロック図である。処理モジュール600は、中間語幹ファイル304から語幹頻度統計を獲得する。処理モジュール600は、ワードコンテンツベクトル学習モジュール200の副構成要素である。処理モジュール600は、中間語幹ファイル304を読み、中間語幹ファイル304中の各文書における各語幹302発生数を計算する。次いで、処理モジュール600は、これらの統計を語幹ハッシュテーブル400中の各データアイテム404の語幹統計セクションに格納する。

【0065】図7は、図2に示すエンティティ選択システムの異なる学習重疊ウィンドウの図である。ワードの同時発生に基づく語幹コンテンツベクトルの初期状態への一連の修正は、語幹ハッシュテーブル400中の語幹コンテンツベクトルのそれぞれについて計算される。本システムは、重疊同時発生ウィンドウを使用して、語幹と同時に発生するワードに基づいて語幹コンテンツベクトルを修正する。図7は、初期ソーステキスト700を示す。いくつかの前処理後、ソーステキスト700は前処理済みテキスト702に変換される。システムは、中心語幹および中心語幹に近似する語幹を解析するため、重疊ウィンドウ704を作成する。このウィンドウは、中心およびその中心の両側に3つまでのワードを有し、これによりウィンドウに合計で7つのワードを提供する。図7に見られるように、ウィンドウ704の中心は、文書の最前部である。それ自体では、ウィンドウは片側式である。語幹「mao」を中心を置く、ウィンドウ704の修正ベクトルは、「rebel」、「group」およびフレーズ「Sendero Luminoso」の初期状態コンテンツベクトルのベクトル関数からなる。語幹「rebel」に中心を置く、ウィンドウ706の修正ベクトルは、「mao」、「group」、「Sendero Luminoso」および「bomb」の初期状態コンテンツベクトルのベクトル関数からなる。ウィンドウは一回につき一つの語幹で移動され、このプロセスは前処理済みテキスト702の各語幹に対して繰り返される。各更新位置において、修正ベクトルが計算され、語幹ハッシュテーブル400（図4に示す）のデータアイテム404中に位置する蓄積修正ベクトルに合計される。

【0066】図8は、近似する語幹のベクトルによって語幹ベクトルを算定する、操作プロセスを例示する図である。語幹ハッシュテーブル400中の各語幹は、近似する語幹のベクトルによって修正される。それぞれの重疊ウィンドウに関して、中心の語幹に関連するベクトル800は、中心語幹に最も近く近似する語幹に関連するベクトル802、804および806によって適応される。

【0067】以下の数式1～4によって、図7に示す重

量ウィンドウにおける各語についての修正を算定する方法を提供する。

$$E_{ij} = N_{ij} - T_j$$

値 E_{ij} は現在の対象（中心） T_j と、近似する N_{ij} との間のエラーおよび補正を表す。ウィンドウ位置におけるすべての現在の近似するものを考慮した T_j への補正

$$C_j^{\text{CURRENT}} = \sum_i^{\text{WS}} (\|E_{ij}\| - \alpha_{ij}) \hat{E}_{ij}$$

【0070】

【0068】
【数1】

(1)

は、数式2および3に示す。

【0069】
【数2】

(2)

【数3】

$$C_j^{\text{CURRENT}} = C_j^{\text{ACCUMULATED}} + C_j^{\text{CURRENT}}$$

(3)

数式2および3に示す変数は、以下のように定義される。

(i) i = 合計仮変数(i i) C_j^{CURRENT} = 語幹 j の補正ベクトル(i i i) $C_j^{\text{ACCUMULATED}}$ = 語幹ハッシュテーブルデータ要素中に格納された語幹 j の実行 (running) 合計補正

(i v) WS = 語幹中のウィンドウサイズ

(v) $\alpha_{ij} = 1 - \Theta_i \Theta_j$ である場合の語幹 i および j の統計による補正効果を制限する近似制限

【0071】

【数4】

$$\Theta_i = B + (1-B) \left(1 - \frac{\log(\frac{1}{ND_i})}{\log(\frac{1}{TND})} \right)$$

数式4は、数式3の変数 α_{ij} を算定するために使用される。数式4に示す変数は以下のように定義される。(i) Θ_i = トレーニングコーパス中の語幹 i の統計的重要性(i i) B = 経験的に決定された領域制御パラメータ(i i i) ND_i = 語幹 i を含む文書の数、および(i v) TND = トレーニングコーパス中の文書の総数

【0072】 以下に含まれる数式5～9は、ベクトルを最終的に更新する。カイド (Ca id) の特許に記載されているアルゴリズムは、一括された更新プロセスを利用することに留意すべきである。このように、そのアルゴリズムは対象（中心）の語幹コンテンツベクトルへの補正を蓄積する。これらの補正是、トレーニングコーパス202を介してのパスの端部においてのみ行われる。

$$T_j^{\text{NEW}} = T_j^{\text{OLD}} + \frac{\gamma}{F_j} C_i^{\text{ACCUMULATED}}$$

【0074】

典型的に、2つのパスのみが卓越した学習結果を達成するためには必要である。数式1～4は、それぞれのウィンドウ位置に適用され、その結果が蓄積される。中間語幹ファイル304中のすべての語幹を通してそれぞれのパスの端部において、数式5～9が適用される。数式5～9に開示されるアルゴリズムは、平均語幹コンテンツベクトルを計算し、これを各正規化された語幹コンテンツベクトルから引いて、語幹コンテンツベクトルの集合全体を再び正規化する。この計算により、語幹コンテンツベクトルが超球の一領域のみに局部化されることを防ぎ、ベクトル空間のよりよい利用を助長する。

【0073】

【数5】

(5)

【数6】

$$T_j^{NEW} = \frac{T_j^{NEW}}{\|T_j^{NEW}\|} \quad (6)$$

【0075】

$$M = \frac{1}{NSTEMS} \sum_i^{NSTEMS} T_i^{NEW} \quad (7)$$

【0076】

$$T_j^{NEW} = T_j^{NEW} - M \quad (8)$$

【0077】

$$T_j^{NEW} = \frac{T_j^{NEW}}{\|T_j^{NEW}\|} \quad (9)$$

数式5から9に使用される変数は以下のように定義される。

- (i) T_j^{OLD} = 旧対象コンテンツベクトル (初期状態または前パス)
- (ii) T_j^{NEW} = 現在のパスの更新された対象コンテンツ
- (iii) α = 学習率
- (iv) F_j = トレーニングコーパス中の語幹 j の例の総数
- (v) $C_j^{ACCUMULATE}$ = このパスについての語幹 j の蓄積された補正ベクトル
- (vi) M = 平均語幹ベクトル
- (vii) $NSTEMS$ = トレーニングコーパス中の固有の語幹の総数

【0078】3. 出力ファイルの書き込み

トレーニングが完了すると、その結果得られた語幹コンテンツベクトルがワードベクトルデータベース208に書き込まれる。

【0079】(セッションコンパイラモジュール) 図9は、セッションコンパイラモジュール210のエンティティベクトル作成プロセスの入力および出力を例示するブロック図である。本発明において、セッションコンパイラモジュール210は、3つの機能を行う。すなわち、テキストからエンティティベクトルへの変形、ファイル形式変換、およびセッション制御ファイル生成である。3つの機能のそれぞれを以下に説明する。

【0080】1. テキストからエンティティベクトルへの変形

データまたは情報の選択を可能にするために、テキストからエンティティベクトル表現への変形が、エンティティベクトルを生成するために必要である。図9は、セッションコンパイラモジュール210がテキストの情報をエンティティベクトルに変換する際、セッションコンパイラモジュール210によって行われる処理を例示する。前処理モジュール1100は、エンティティ記述データベース216からエンティティ記述を読む。前処理モジュール1100は、図2に示すセッションコンパイラモジュール210によって行われるプロセスの例である。

ラモジュール210の副構成要素である。前処理モジュール1100は、図3に示す前処理モジュール300の一部として行われる前処理と同様である。前処理モジュール1100は、エンティティ記述データベース216から読まれた現在のワードがストップリストファイル204またはフレーズリストファイル206にあるかどうかを判定する。次いで、有効な語幹302は、エンティティベクトル形成モジュール1102によって使用されて、スタティックエンティティベクトルを生成し、次いでこのスタティックエンティティベクトルはスタティックコンテンツベクトルデータベース234に格納される。

【0081】数式10は、エンティティベクトルを形成するため、図2に示すセッションコンパイラモジュール210によって行われるプロセスの例である。エンティティのコンテンツベクトルは、そのエンティティを含むシンボルの正規化加重和である。テキストの記述であるこれらのエンティティに関して、記述ベクトルはその記述中の語幹の語幹コンテンツベクトルの正規化加重和である。

【0082】

【数10】

$$ECV_i = \frac{\sum_{j=1}^{NSTEMS_i} \log\left(\frac{TND_j}{ND_j}\right) SCV_j}{\left| \sum_{j=1}^{NSTEMS_i} \log\left(\frac{TND_j}{ND_j}\right) SCV_j \right|} \quad (10)$$

式10の変数は以下のように定義される。

- (i) ECV_i =エンティティ*i*のエンティティコンテントベクトル
- (ii) SCV_j =語幹*j*の語幹コンテンツベクトル
- (iii) TND =トレーニングコーパス中の文書の総数
- (iv) ND_j =語幹*j*を含むトレーニングコーパス中の文書数
- (v) $NSTEMS_i$ =エンティティ*i*の記述中の語幹数

【0083】2. ファイル形式変換

図10aは、図2に示す高速ファイル232を作成するプロセスを例示するブロック図である。セッションコンパイラモジュール210は、ワードコンテンツベクトル学習中に計算するのに効率のよいデータ構造形式は、エンティティ選択中にRTASモジュール214による使用には不適当であるため、エンティティ選択プロセスを容易にするようファイル形式変換を行う。ワードコンテンツベクトル学習プロセスに必要な動作の柔軟性および複雑性はRTASモジュール214には必要ではなく、そのため相当な簡略化が行われる。この変形の一部として、語幹コンテンツベクトルのすべては、トレーニングコーパス202での発生頻度によって前もって計られる。語幹コンテンツベクトルは、最終的な正規化ステップなしで等式10に示すものと同じ因数によって計られる。さらに、低頻度の語幹（トレーニングコーパスにおいて4回未満発生するもの）は、ファイルから除去される。残りの語幹は語幹集団全体の統計的に最も有意な部分を表している。次いでセッションコンパイラモジュール210は、このように切頭され、かつ前もって計られた結果として得られた語幹コンテンツベクトルを高速ファイル232に書き込む。

【0084】3. セッション制御ファイル生成

図10bは、セッション制御ファイル215を生成するプロセスを例示するブロック図である。セッションコンパイラモジュール210の主な機能は、セッション記述ファイル215に含まれるオペレータが作成したセッション記述からセッション制御ファイル215を生成することである。セッション制御ファイル215は、広告選択プロセス中にRTASモジュール214をガイドするため、機械読み取り可能な形式の命令を含む。セッション制御ファイル215は、セッション記述ファイル212において特定された情報および初期エンティティベクトルのセットを含む。セッション制御ファイル215

は、選択に適格な広告のグループを識別し、EVUXモジュール228が動作可能ではない場合に、バックアップエンティティベクトルのセットを提供する能力を有する。セッション記述ファイル212中のセッション情報は、選択がアクティブとなるのに適格な広告等のエンティティのセットを記述する。セッション制御ファイル215は、各広告について、以下の属性に関する制御メカニズムを含む。

- ・スタティックまたは動的属性制御。広告はコンテンツベクトル空間中の固定位置にスタティックに置かれるか、あるいはユーザの応答に基づいて動的に適応されるのを可能にする。

- ・リセット。リセットの機能性が特定されている場合、既存の広告のエンティティベクトルは初期ベクトルにリセットされる。

- ・対象通しカウント。対象通しカウントは、広告に配信されるべき通し数を決定する。

- ・開始日。開始日は、広告が選ばれる資格があるようになる日と時間を決定する。

- ・終了日。終了日はその広告が選ばれる資格のなくなる日と時間を決定する。

- ・初期状態。広告者は、その広告についての初期エンティティベクトル選択を記述することができる。初期状態選択には、以下のオプションが含まれる。

- (i) 広告および前述のテキストからベクトルへの変換プロセスに基づくベクトル

- (ii) 特定の広告のエンティティベクトルに基づくベクトル

- (iii) 外部ファイルに含まれるエンティティベクトルに基づくベクトル

【0085】また、セッションコンパイラモジュール210は、すべてのアクティブなエンティティ情報の保管場所であるEVUSモジュール228（図2に示す）との接続を確立する。初期エンティティベクトルは、広告について算定され、広告識別子、パラメータ及びベクトルはエンティティベクトルデータベース230に格納されるため、EVUSモジュール225に伝送される。

【0086】（リアルタイム広告サーバモジュール）RTASモジュール214は、様々な広告管理ルーティンを行うために、インターフェースのアプリケーションプログラム(API)局部的または遠隔ソフトウェアアプリケーション呼び出しを提供する。RTASモジュール214は、セッション制御ファイル215中にセットされたガイドラインに従ってユーザに表示する広告を選択する。

【0087】広告選択プロセス中、RTASモジュール214は、以下にリストするプロセスステップを行う。

【0088】1. 初期化

RTASモジュール214が初期化コマンドを受け取ると、RTASモジュール214は、セッション制御ファイル215を読み、メモリに格納されている内部制御テーブルを初期化する。次いで、RTASモジュール214は、セッション制御ファイル215の内容によって特定された、適切なエンティティコンテントベクトルを要求する。RTASモジュール214は、EVUSモジュール228を介してエンティティベクトルデータベース230からアクティブなエンティティについてのエンティティベクトルを要求し、RTASモジュール214内のメモリ常駐データ構造にそれらを格納する。初期化が完了すると、RTASモジュール214はリアルタイム動作の準備ができたことになる。

【0089】2. ユーザ行動ベクトルの決定

ユーザ動作メッセージがクライアントAPIを介してシステムに入ると、RTASモジュール214は、そのユーザについての行動ベクトルを確立する。ユーザがすでにシステムに知られている場合、ユーザのウェブブラウザが提供するユーザIDがデータベース検索の鍵として使用される。RTASモジュール214は、ユーザIDをPVUSモジュール224に渡す。次いで、PVUSモジュール224はプロフィールベクトルデータベース226からそのユーザに関連するプロフィールベクトルを戻す。ユーザがシステムに知られていない場合、新しい初期化されたベクトルが戻される。いずれの場合においても、RTASモジュール214は後続する動作に有効なユーザ行動ベクトルを有する。このベクトルはRTASモジュール214内の内部メモリテーブル内に格納される。

【0090】3. ユーザ行動の観察

RTASモジュール214は、また、ユーザ行動をユーザのプロフィールベクトルに組み込む。これらの行動には、図2に示すように、ウェブページ要求218、または探索照会217が含まれる。クライアントコンピュータ219は、クライアントAPI中の観察された行動を含む行動パケットを渡す。RTASモジュール214は、ユーザ行動パケットに含まれる情報を用いて、そのユーザ動作に対する行動ベクトルを生成する。

【0091】図11は、図2に示すRTASモジュール214が行う、ページ要求をコンテンツベクトルに変形するプロセスの入力と出力を例示するブロック図である。観察されたユーザ行動がページ、ナビゲーション、または階層要求218を含む場合、RTASモジュール214の一部であるページコンテンツベクトル検索モジュール1402は、要求されたページのIDに関連するエンティティベクトルをスタティックコンテンツベクトルデータベース234から検索する。RTASモジ

ル214は、行動ベクトルを提供するために、ユーザが要求したページのエンティティベクトルモジュールをユーザIDによって増大させる。

【0092】図12は、図2に示すリアルタイム広告サーバモジュールによって行われる、照会をコンテンツベクトルに変形するプロセスの入力と出力を例示するブロック図である。クライアントコンピュータ219がユーザ照会217等のユーザの行動を観察する場合、RTASモジュール214は照会用語に基づいたエンティティベクトルを提供するため、高速ファイル232のメモリイメージを使用することによって、照会217をコンテンツベクトルに変形する。RTASモジュール214の一部である前処理モジュール1502は、ストップリストファイル204において識別された共通のワードを取り除き、フレーズリストファイル206の一部であるワードを組み合わせ、ユーザ照会217中の用語すべてについて語幹化プロセスを行う。次いでRTASモジュール214は、形式エンティティコンテンツベクトルモジュール1504が前述した照会217の結果をそのユーザIDと共に増大させることによって、ユーザ行動ベクトルを作成する。

【0093】4. 提示するのに最もよいエンティティの選択

図13は、図2に示すRTASモジュール214によって行われる広告選択プロセスの機能を例示するブロック図である。クライアントコンピュータ219(図2に示す)がユーザ行動を観察しない場合、RTASモジュール214は、提示するのに最もよいエンティティを選択するためのベースとして、プロフィールベクトルデータベース226からの履歴ユーザプロフィールベクトルを用いる。クライアントコンピュータ219が対話型行動を観察する場合、RTASモジュール214は、その観察されたユーザ行動を、提示するのに最もよいエンティティを選択するためのベースとして用いる。

【0094】RTASモジュール214は、履歴ユーザプロフィールベクトルまたは行動ベクトルを、セッション制御ファイル215によって特定されるすべてのアクティブな広告に対するベクトル比較のベースとして用いる。セッション制御ファイル215の内容は、初期化中にRTASモジュール214によってメモリ中にロードされ、かつ高速探索に利用可能である。セッション制御ファイルをRTASモジュール214のメモリに置くことによって、性能利得を実現する。

【0095】再度図13を参照して、RTASモジュール214は、行動およびユーザベクトルでエンティティベクトルの点乗積を行うことによって、最適なエンティティを選択する(状態1600)。RTASモジュール214は、比較するベクトルとすべてのアクティブ広告ベクトルとの間のベクトル近似の距離を計算することによって、探索プロセスを行う。RTASモジュール21

4は任意のベクトル近似計量を用いることができるが、本発明の好ましい実施形態では、ベクトルの内積を使用する。

【0096】状態1602で、RTASモジュール214は、在庫管理制限の制約を受ける最も近い広告を選択する。具体的に、最も近い広告がすでにユーザの集合に十分な回数提示されている場合、RTASモジュール214は選択された二番目に最も近くマッチした広告を提示する。在庫管理スキームは、任意に複雑であってもよく、外部在庫管理モジュールとインタフェースする規定を有し、これを一連の外部からの制約を受ける広告提示に適応してバランスを取るために使用してもよい。RTASモジュール214の副構成要素である在庫加重モジュールは、ベクトル近似ウェートを調整することによって、すべての在庫制御動作を行う。次いで、RTASモジュール214は、スコアで各広告IDを測定し、ランク付けする（状態1604）。

【0097】上述した制約を受ける最近似エンティティすなわち広告が選択されると、RTASモジュール214はAPIを介して選択した広告のIDをクライアントコンピュータ219に渡す。そのIDは、ウェブを介して実際の広告画像を引き出し、ユーザに表示するため、クライアントコンピュータ219によって使用される。

【0098】性能向上させるために、RTASモジュール214はアクティブセッション制御ファイル215によって記述されたエンティティベクトルおよびパラメータのセットを一時保存する。RTASモジュール214は、変更されたエンティティのためにEVUSモジュール228から定期的に更新される。

【0099】5. エンティティ提示に対するユーザの応答の観察

最もよいエンティティすなわち広告が決定され、ホストシステムがその選択された広告を組み込んだウェブページを生成すると、ユーザのその広告に対する応答が監視される。エンティティすなわち広告上をクリックすることで示されるように、ユーザがその製品についてより多くの情報を要求した場合、この事象はAPIを介してRTASモジュール214に送信される。しかし、すべての広告提示が必ずしもRTAS214モジュールにメッセージを送信する必要がある結果とはならない。

【0100】6. ユーザ行動ベクトルの更新
RTASモジュール214は、観察されたユーザ動作を反映するように、ユーザ行動プロフィールベクトルを適応する。次いで、観察されたユーザ行動ベクトルを使用して、そのユーザについてのプロフィールベクトルを更新する。RTASモジュール214は、履歴ユーザプロフィールベクトルを修正するため、そのユーザIDおよび観察されたユーザ行動ベクトルをPVUSモジュール224に送信する。PVUSモジュール224の動作を以下に詳述する。

【0101】7. 広告すなわちエンティティベクトルの更新

ユーザがエンティティ（広告、クーポンまたは勧誘）に応答した場合、そのエンティティベクトルおよびユーザIDが更新のためにEVUSモジュール228に送信される（図17に示す）。次いで、EVUSモジュール228は、エンティティベクトルをユーザの観察された行動ベクトルに向けて適応させる。EVUSモジュール228の動作は以下に詳述する。

【0102】（プロフィールベクトル更新サーバモジュール）PVUSモジュール224は、プロフィールベクトルデータベース224に含まれるユーザプロフィールベクトルの格納、検索、および更新に対しての責任がある。

【0103】これらのユーザプロフィールベクトルは、以下を含む広範囲の処理能力のベースである。

(i) 対象の広告、クーポン、申し出および情報をユーザに広める。

(ii) ユーザ行動の解析

(iii) ユーザ行動と広告、製品およびサービス記述の相関関係

(iv) ユーザプロフィールベクトルと人口学的購入データベースとの相互関係に基づくユーザ特徴の推論

(v) 経時的にユーザ行動の追跡

(vi) 起こりうるユーザ行動の予測モデル化およびこれらの予測モデルに基づく情報提示の最適化

(vii) 予測モデルを最適化し、管理報告書を提供し、広告収入の増加への基礎を準備するため、有効性およびクリックスルー統計の蓄積

【0104】本発明の好ましい実施形態では、PVUSモジュール224は、クレジットカードによる購入履歴等の外部情報もユーザプロフィールに組み込む。コンテンツベクトル表示がウェブ行動およびクレジットカード購入等の異種の情報を均一化された（unified）高次元空間に収容しているため、この情報を容易に組み込むことができる。

【0105】PVUSモジュール224は、TCP/IP等のネットワークプロトコル上でクライアントから要求を受信する、一つ以上の実行可能なコンピュータプログラムとして実施され、このような要求に対して応答を生成する。PVUSモジュール224は、一つ以上の物理的なデータベースを有し、これは固有のユーザIDによって索引を付けられたユーザプロフィールコンテンツベクトルおよびパラメータを格納するために使用される。多PVUSモジュールが、多数のユーザを扱うために同時に動作する場合もある。多サーバが使用される場合、それらはクライアントによって一つのエンティティとして対照となる。PVUSモジュールは、大きなユーザ集団をサポートするため、多ハードウェアプラットフォームにわたって計られる場合もある。この分布は、ユ

ユーザIDをN個の物理的なサーバのうちの選択された一つに多重送信することによって果たされる。各サーバはさらにその要求をM個の物理的なデータベースの一つに多重送信する。多重送信は、ユーザIDのハッシュコードを生成することによって果たされ、結果として主にハッシュコード中にランダムビット分布を生じさせる。使用することのできる多数の多重送信ハッシング法がある。しかしながら、好ましい実施形態では、MD5アルゴリズムを使用する。次いで、マルチブレクサはハッシュコード上でモジュロNを使用して、N個のチルドレンのいずれがその要求を取り扱うべきかを決定する。

【0106】PVUSモジュール224は以下に列挙する処理動作を行う。

【0107】1. 新規ユーザプロフィールベクトルの作成

システムにおけるユーザにはそれぞれ、固有のユーザIDが割り当てられる。ウェブ環境においては、これはユーザのインターネットのIPアドレスか、あるいはブラウザの「クッキー」ファイルに置かれているホストが選択したIDであってもよい。プロフィールベクトルデータベース226に含まれていないユーザIDを作るよう要求が行われた場合、PVUSモジュール224は新規のユーザについてのエントリを作成し、新規ユーザプロフィールベクトルを、後に適応アルゴリズムによって調整される（図19aおよび図19bに示す）擬似乱数コンテンツベクトルに初期化する。

【0108】2. ユーザプロフィールベクトルおよびパラメータの検索

PVUSモジュール224は、固有のユーザIDを使用してユーザプロフィールベクトルを得て、ハッシュコードを決定する。その位置でのハッシュテーブルにおけるエントリは、プロフィールベクトルデータベース226中の情報へのポインタを含む。PVUSモジュール224は、そのデータベースのポインタを使用して、ユーザプロフィールベクトルの検索を行う。プロフィールベクトルの要求がRTASモジュール214からである場合には、PVUSモジュール226は次の使用のために、そのプロフィールベクトルをRTASモジュール214に渡す。

【0109】3. ユーザプロフィールベクトルの適応
PVUSモジュール224は、ユーザの行動についてのいくつかの観察に基づいて、ユーザプロフィールベクトルの適応を行う。ユーザが本発明を利用しているウェブサイトを横切る際、ユーザが要求した照会や、ウェブページおよび広告を含む行動が観察される。観察された行動のそれぞれは、コンテンツベクトルに変換され、プロフィールベクトルデータベース226に格納されているユーザのプロフィールベクトルを適応させるのに使用される。

【0110】図14は、図2に示すPVUSモジュール

224のプロフィールベクトル適応プロセスを例示するブロック図である。PVUSモジュール224は、コンテンツベクトルへの変形照会(TQCV)モジュール1700、動作をコンテンツベクトルに変形する(TACV)モジュール1704、プロフィールベクトルルックアップ(PVL)モジュール1706およびプロフィールベクトル適応(PVA)モジュール1702を含むいくつかの副構成要素を含む。ユーザが照会217をシステムに入力した場合、照会ストリングおよびそのユーザIDがTQCVモジュール1700に渡される。TQCVモジュール1700はその照会ストリングを、図10bに示すプロセスと同様に照会コンテンツベクトルに変形する。結果として得られた照会コンテンツベクトルは、ユーザプロフィールベクトルを適応するために、PVAモジュール1702に渡される。PVAモジュール1702は、以下に説明され、図19aおよび図19bに示される適応アルゴリズムを利用する。プロフィールベクトルデータベース更新モジュール1704は、後の使用のために、更新されたプロフィールベクトルをプロフィールベクトルデータベース226に書き込む。

【0111】ユーザが広告をクリックしたり、または製品を購入222すると、TACVモジュール1704がその広告または購入記述を使用して、ユーザプロフィールベクトルを適応する。このモジュール1704は、PVUSモジュール224内でメモリにあるテーブルルックアップを介して、エンティティIDをコンテンツベクトルに変形する。平行して、ユーザIDは、プロフィールベクトルデータベース226からそのユーザのプロフィールベクトルをルックアップするために使用される。PVAモジュール1702はこれら2つのベクトルを図19aおよび図19bに記載される適応アルゴリズムに入力する。次いで、プロフィールベクトルデータベース更新モジュール1704は、後の使用のために、その更新されたプロフィールベクトルをプロフィールベクトルデータベース226に書き込む。

【0112】4. 人口学的予測およびタグ付け

多くの広告およびセールスアプリケーションのために、全体のユーザ集団と同様に個人についての人口学的パラメータの推定を有することは、有利である。本システムにおいて、ユーザ登録は行われないが、広告者はプロフィールベクトルを解析して、ユーザ人口を検索する。広告者は、これらのプロフィールベクトルを使用して、ユーザの年齢、性別、世帯収入(HHI)および他の人口学的パラメータを推定することができる。本システムは、その動作に人口学的モデルを組み込むよう用意がしてある。これらの人口学的モデルを使用する場合、ユーザの集団全体の人口学的パラメータを推定し、使用して、プロフィールベクトルデータベース226中でその情報を増加させる。この推定プロセスは、従来の統計方法の両方あるいはニューラル・ネットワーク技術を用い

ることによって行うことができる。

【0113】図15は、図2に示す関係、発見および解析モジュールによって行われるモデル構成プロセスの入力および出力を例示するブロック図である。年齢、性別およびHHIの予測モデルは、履歴ユーザ動作データベース226中に含まれる以前に観察された行動、異なるサービスでのユーザ登録からのグラウンド・トゥルース情報あるいはグラウンド・トゥルース・ユーザ人口学的データベース1904を介して、およびプロフィールベクトルデータベース226からのプロフィールベクトルの組み合わせを使用して作成される。この情報を使用して、モデル化モジュール1900は、モデルパラメータのセットを作成する。これらのモデルパラメータはモデルパラメータデータベース1906に格納される。予測モデルパラメータを導出するために、ニューラル・ネットワーク技術を含むあらゆる統計的に有効なプロセスをモデル化モジュール1900において使用することができる。予測モデルの典型的な出力は年齢、性別およびHHIである。

【0114】図16は、PVUSモジュール224が人口学的情報でプロフィールベクトルデータベース226を更新するプロセスを例示するブロック図である。プロフィールベクトルデータベース中の各ユーザについて、PVLモジュール1706はプロフィールベクトルデータベース226からそのユーザに関連するプロフィールベクトルを検索する。PVLモジュール1706はこの情報をユーザIDおよび履歴動作と共に予測モデル処理モジュール2000に渡す。予測モデル処理モジュール2000は、現在のユーザプロフィールベクトルおよび以前観察された行動と共に、モデルパラメータデータベース1906からのモデルモデルパラメータを使用して、現在のユーザについての人口学的パラメータのセットの推定を形成する。この推定は、プロフィールベクトル更新モジュール1704によってプロフィールベクトルデータベース226に含まれる情報に追加される。増加された内容は、広告選択中にRTASモジュール214への入力として、および関係発見および解析プロセスの一部として使用される。

【0115】(エンティティベクトル更新サーバモジュール)EVUSモジュール228は、エンティティベクトルデータベース230からのエンティティプロフィールベクトルの格納、検索および更新に責任がある。EVUSモジュール228は、エンティティベクトルデータベース230を読み、および書き込むための、システムにおける主要な処理モジュールである。

【0116】EVUSモジュール228は、ネットワークプロトコル(TCP/IP)上でクライアントからの要求を受信し、そのような要求に対する応答を生成する一つ以上の実行可能なコンピュータプログラムからなる。EVUSモジュール228は、固有のエンティティ

IDがユーザプロファイルコンテンツベクトルおよびパラメータを効果的に格納および索引付けるのに使用する一つ以上の物理的なデータベースを有してもよい。

【0117】正常動作中に、EVUSモジュール228は以下の処理を行う。

【0118】1. エンティティ挿入および削除

EVUSモジュール228は、セッションコンパイラモジュール210のコマンドに応答して、新しいエンティティすなわち広告を挿入し、古いエンティティすなわち広告をエンティティベクトルデータベース230から取り除く。新しいセッションがコンパイルされると、セッションコンパイラモジュール210はEVUSモジュール228に更新コマンド、新しいエンティティベクトルおよびパラメータを送信する。さらに、セッションコンパイラモジュール210は、エンティティベクトルデータベース230から非アクティブなエンティティを削除するよう、EVUSモジュール228に対してコマンドを送信する。

【0119】2. エンティティベクトルおよびパラメータの格納および検索

EVUSモジュール228は、システム中のすべてのアクティブなエンティティについての現在のパラメータおよびベクトルを格納および検索し、所与のエンティティIDについてのエンティティベクトルおよび関連パラメータを検索する。RTASモジュール214の初期化中、RTASモジュール214は、RTASモジュール214のメモリに保持するために、より最近のエンティティベクトルおよびパラメータのコピーを要求するコマンドを、EVUSモジュール228に送信する。

【0120】3. エンティティプロフィールベクトルの適応

図17は、図2に示すEVUSモジュール228のエンティティベクトル適応プロセスを例示するブロック図である。EVUSモジュール228は、ユーザがエンティティに応答(エンティティをクリック)したことを示すメッセージに応答して、エンティティベクトル適応を行う。EVUSモジュール228は、エンティティベクトル適応モジュール2100、コンテンツベクトルへの変形照会(TQCV)モジュール2102、エンティティベクトルルックアップモジュール2104、プロフィールベクトルルックアップモジュール2106およびエンティティベクトルデータベース更新モジュール2108を含む様々な副構成要素を含む。

【0121】適応事象は、ユーザが照会217を入力するか、広告をクリックする／あるいは製品222をオンラインで購入したときに発生する。エンティティ適応事象が発生すると、即ちユーザ照会217等の事象を引き起こしたユーザ動作が生じると、EVUSモジュール228は、RTASモジュール214から選択された広告およびそのユーザのIDを得る。

【0122】ユーザが照会217を要求した場合、TQCVモジュール2102はその動作を照会コンテンツベクトルに変形し、その結果をエンティティベクトル適応モジュール2100に渡す。ユーザ事象がクリックスルーまたは購入222である場合には、エンティティベクトルルックアップモジュール2104がエンティティベクトルデータベース230から選択されたエンティティのエンティティベクトルを検索する。エンティティベクトルルックアップモジュール2104は、選択されたエンティティのベクトルをエンティティベクトル適応モジュール2100に渡す。ユーザIDがEVUSモジュール228に渡された場合には、プロフィールベクトルルックアップモジュール2106がプロフィールベクトルデータベース226からそのユーザのプロフィールベクトルを検索する。その結果得られたコンテンツベクトルは、次にエンティティベクトル適応モジュール2100に渡される。

【0123】TQCVモジュール2102から入力を受信した後、エンティティベクトルルックアップモジュール2104、プロフィールベクトルルックアップモジュール2106およびエンティティベクトル適応モジュール2100は、図19Aおよび図19Bに示す数式を行う。次いで、エンティティベクトル適応モジュール2100は、その更新されたエンティティベクトルをそのエンティティIDおよびタイムスタンプと共に、その情報がエンティティベクトルデータベース230に格納されるエンティティベクトルデータベース更新モジュール2108に渡す。

【0124】4. 時間更新

EVUSモジュール228は、ベクトルまたはパラメータが所与の時間以来変更されているエンティティIDのリストを生成する。この情報は、RTASモジュール214が維持するEVUSモジュール228のメモリ画像をリフレッシュするために使用される。最新の更新のタイムスタンプを含むリフレッシュコマンドがRTASモジュール214から送信されると、EVUSモジュール228は、エンティティID、変更されたアイテムのベクトルパラメータおよびタイムスタンプをRTASモジュール214に送信する。次いで、この情報はエンティティ選択プロセスに使用するために、RTAS内部メモリに保存される。

【0125】5. 在庫管理

EVUSモジュール228は、システムにおけるすべての広告にわたって在庫管理加重を行う。EVUSモジュール228は広告選択スコアのそれぞれを調整するため、ウェートパラメータを使用する。ウェートパラメータは、広告対象付けられた通し数および各広告の時間残量によって決定される。

【0126】(プロフィールおよびエンティティ適応アルゴリズム) 図18は、ユーザ照会217(図2に示

す)の受信後の、プロフィールベクトル更新サーバおよびエンティティベクトル更新サーバのベクトル適応プロセスを例示するフローチャートである。PVUSモジュール224およびEVUSモジュール228の両方は、システムとのユーザの対話の結果としてのプロフィールベクトルおよびエンティティベクトルを進化させるため、適応アルゴリズムのセットを適用する。

【0127】状態2250で始まり、PVUSモジュール224は現在のユーザ照会217に基づいてプロフィールベクトルを更新する。決定状態2252に移り、ユーザのプロフィールベクトルおよび探索照会の行動ベクトルがエンティティベクトルと比較され、スコアを確立するために、この比較に従ってランク付けする。ユーザのプロフィールスコアは、探索照会の行動スコアに対して比較され、どちらのスコアが高いかを決定する。照会スコアのほうが高い場合、PVUSモジュール224はバイアスペクトルを更新(状態2254)するよう進む。このバイアスペクトルは、連続的な相互適応プロセス中すべてのベクトルが同じエンティティベクトルにならないことを確実にするために使用される。次に、状態2258で、広告が照会スコアに基づいて選択される。

【0128】決定状態2260に進み、システムはユーザが提供された広告をクリックしたかどうかをチェックする。ユーザが広告を選択しなかった場合、システムは状態2250に戻り、将来のユーザ照会に対して適応プロセスを繰り返す。ユーザが広告を選択した場合、EVUSモジュール228がユーザ照会217の行動ベクトルで選択された広告のベクトルを更新する(状態2262)。次に、状態2264において、PVUSモジュール224がその広告のエンティティベクトルでプロフィールベクトルを更新し、状態2274に移る。

【0129】再び決定状態2252を参照して、プロフィールスコアが照会スコアよりも高い場合、状態2265においてバイアスペクトルが更新される。次に、状態2266において、プロフィールスコア付けによって選択された広告がクライアント219に送信される。決定状態2268に進み、システムは、ユーザが提供された広告をクリックしたかどうかをチェックする。ユーザが広告を選択しなかった場合、システムは状態2250に戻り、その後にユーザ照会を受信すると適応プロセスを繰り返す。ユーザが広告を選択した場合、PVUSモジュール224がその広告のエンティティベクトルでプロフィールベクトルを更新する(状態2270)。次いで、EVUSモジュール228がそのユーザ照会217のプロフィールベクトルで、選択された広告のベクトルを更新する。

【0130】状態2264または状態2274から移り、状態2274において、選択された広告に関するコストモデルが更新される。次いでシステムは、状態2250に戻り、その後にユーザ照会を受信すると適応プロセスを繰り返す。

ロセスを繰り返す。

【0131】個人ユーザのプロフィールおよび対応する現在の照会は、数式11および12に示す2つのセット

$$P_{uid} = \{p_1^{(uid)}, \dots, p_d^{(uid)}\}$$

【0133】

$$Q_{uid} = \{q_1^{(uid)}, \dots, q_e^{(uid)}\}$$

【0134】ユーザ照会は、一つ以上のコンテンツベクトルによって表される場合もある項目別情報を含む。数式13は、各広告がプロフィールベクトルと同様な方法

$$A_{add} = \{a_1^{(add)}, \dots, a_f^{(add)}\}$$

【0136】数式14および数式15は、図15に示す決定状態2252においてスコアを計算する際適用される。

$$p_score(P_{uid}, A_1, \dots, A_n) = \max_{i,j,add} [p_i \cdot a_j^{(add)} Value_{add}] \quad (14)$$

【0138】

$$q_score(Q_{uid}, A_1, \dots, A_n) = \max_{i,j,add} [q_i \cdot a_j^{(add)} Value_{add} \times \mathbb{1}(q_i \cdot a_j^{(add)} > threshold_{query})] \quad (15)$$

「Value」パラメータは従来のコストモデルアルゴリズム（図示せず）によって計算される。

【0139】図19aおよび図19bは、図2に示すプロフィールベクトル更新モジュールおよびエンティティベクトル更新モジュールによって行われる一定率の学習アルゴリズムの数式流れ図である。別の方法で注記されない限り、図19aおよび図19bに示す数式はプロフィールおよびエンティティコンテンツベクトル更新の両方に共通する。適応プロセスを制御するために使用するパラメータは、図20に示す。

【0140】以下は、図18の状態に関連して、図19aおよび図19bに例示する数式がいつ行われるかを説明している。ブロック2276、2284（図19aに示す）の数式は、状態2250（図18に示す）において、プロフィールベクトルを照会で更新するプロセス中に行われる。ブロック2278の数式は状態2262において、ユーザが広告を選択した後に広告ベクトルをユーザ照会で更新するプロセス中に行われる。同様に、ブロック2280の数式は、状態2264において、選択された広告でプロフィールベクトルを更新する間に行われる。

【0141】ブロック2282（図19aに示す）の数式は、状態2254（図18に示す）においてすべてのベクトルが同じ多次元ベクトルにならないことを確実にするためにバイアスペクトルを更新する間に行われる。ブロック2286の数式は、状態2272において、広告ベクトルをプロフィールベクトルで更新する間に行われる。これらの数式は、ユーザのプロフィールスコアが照会スコアよりも高い場合に行われる。同様に、ブロッ

によって表される。

【0132】

【数11】

(11)

【数12】

(12)

で表されることを示す。

【0135】

【数13】

(13)

【0137】

【数14】

(14)

【数15】

ク2288（図19bに示す）の数式は、状態2270（図18に示す）において、プロフィールベクトルを広告ベクトルで更新する間に行われる。

【0142】ブロック2290（図19bに示す）の数式は、状態2265（図18に示す）において、ユーザのプロフィールスコアがユーザ照会のスコアよりも高くなった後、バイアスペクトルを更新する間に行われる。

【0143】（エンティティ在庫加重（コストモデル）の算定）RTASモジュール214は各エンティティすなわち広告について生成された通し数を追跡し、広告IDおよび各エンティティすなわち広告について生成された通し数を含むEVUSモジュール228にメッセージを定期的に送信する。RTASモジュール214は、通しの対象数、および各広告が在庫管理ウェートを算定するよう機能する資格を有した状態の時間を特定する広告パラメータを使用する。ウェートは、アクティブ広告をスコア付けするときに、RTASモジュール214によって適用される。RTASモジュール214によって、押し数の少ない広告がより多くの通しを受け、一方、押し数が過分になされた広告では通し数を抑制する。

【0144】（関係発見および解析処理モジュール）本システムは、プロフィールベクトルデータベース226およびエンティティベクトルデータベース230（図2に示す）のコンテンツにおいて、関係発見および解析を行う能力を有する。解析プロセスの結果はアーリスト240に提示される一連の報告書に列挙される。これらの報告書はシステムユーザの行動パターンについての価値の高い動作可能な情報を提供する。

【0145】RDAモジュール236は、プロフィール

ベクトルデータベース226およびエンティティベクトルデータベース230の内容を探って、解析報告書238（図2に示す）を作成する。この解析を行うために、様々な従来の統計技術またはニューラル・ネットワークアプローチが用いられる。

【0146】RDAモジュール266は、非管理分類、管理分類および従来の統計学を含む3つの解析的な機能を行う。3つの機能のそれぞれを以下に説明する。

【0147】1. 非管理分類

RDAモジュール236は、プロフィールベクトルデータベース225、エンティティベクトルデータベース230またはこれらモジュールの両方の内容の非管理分類またはクラスタ化を行う。RDAモジュール236は、アルゴリズムが実数ベクトルを処理できる限り、いずれのクラスタ化アルゴリズムを用いてもよい。クラスタ化の結果は、高密度領域を表すクラスタ重心ベクトルのセットである。これらの重心は、他のすべての処理動作に使用できる同一タイプのベクトルである。したがって、RDAシステムは、コンテンツベクトル空間中で重心ベクトルに最も近いワードベクトルのセットを見つけることができる。これらのベクトルに関連するワードおよびそれらの近似距離は重心の主要な概念を形成する。経時のクラスタ重心の発展は、傾向情報を提供する。これらのグラスタ中心に近いワードベクトルの識別は、各クラスタについてのキーワードの要約を提供し、解析および解釈タスクを容易にする。また、従来の統計解析も、クラスタのコンテンツを検査するために使用することができる。クラスタメンバーシップの分散は、クラスタコヒーレンスおよび行動の点では「焦点の均一性」の信頼性のある指標である。製品ベクトルデータベース226において識別されたクラスタはその製品を購入したユーザの特徴について非常に価値ある情報を提供する。この結果は、広告ベクトルとユーザコンテンツベクトルとの間の相互適応の当然の結果である。見つけられたクラスタの数およびクラスタの説明は、アリスト240による検閲のために、解析報告書238にリストされる。

【0148】2. 管理分類

RDAモジュール236もまた、プロフィールベクトルデータベース226、エンティティベクトルデータベース230またはこれらモジュールの両方の内容の管理分類を行う。管理分類のために、RDAモジュール236は、アルゴリズムが実数ベクトルを処理できるいずれのアルゴリズムを用いてもよい。分類の結果は、各分類のメンバーである、ユーザおよび／またはエンティティのセットである。分類によるカウントおよびメンバーシップは、アリスト240による検閲のために、解析報告書238にリストされる。

【0149】3. 従来の統計学

RDAモジュール236は、プロフィールベクトルデータベース226、エンティティベクトルデータベース2

30またはこれらモジュールの両方の内容に従来の統計学を適応する。この解析は、カウント、カウント分布、データベースの内容全体またはアリスト240が選択した任意の部分セットの平均および分散を生成する。処理結果は、アリスト240による検閲のために、解析報告書238にリストされる。

【0150】広告シナリオについての本発明の動作の概要を以下に示す。

【0151】（システム初期化）まず、ワードコンテンツベクトル学習モジュール200がテキスト入力から多コンテンツベクトルを作成する。この技術の結果、ワードコンテンツベクトルのセットが得られ、1つのコンテンツベクトルはトレーニング語彙中に見られ、ストップリスト204には見られないワードのそれぞれについてのものである。広告シナリオでは、通常トレーニングテキストは製品の説明または業界紙である。たとえば、動作の対象領域がスポーツ製品である場合、トレーニングコーパスはスポーツ製品カタログからの製品の説明、

「Field and Stream」誌の数年間の内容およびアウトドアライフの本を含むだろう。アウトドアスポーツ等の動作の領域に適応させるのに正確なトレーニングコーパスの選択は、ワードの使用に正確な「スピル」を与える。上記の例において、「弓」は、「バイオリン」または演技の終了時にステージで行われる動作（お辞儀を意味する）とは対照的に、「矢」と関連づけるようになる。

【0152】次に、セッションコンパイラモジュール210が販売製品の説明からエンティティベクトルの初期セットを計算する。これらの説明は、製品カタログからのものでもよいし、またオペレータの人間がタイプしてもよい。たとえば、「暖かいゴアテックス、ゲレンデでの凍てつく冬の昼夜に最適な、ダウンパーカー」は、コートのエンティティベクトルのためのテキスト初期化として役立つであろう。エンティティベクトルは、説明におけるワードのワードベクトルの正規化加重和である。セッションコンパイラモジュール210は、システム中の各広告についての初期エンティティベクトルを作成する。

【0153】セッション記述ファイル212は、認可された表示回数、サイト、どの広告を表示するか、およびいくつの通しが利用可能か等の広告設置のガイドラインを記述し、機械が利用可能な形式にコンパイルされる。この結果、広告提供（serving）プロセスの動作を制御する情報を有するセッション制御ファイル215の作成がもたらされる。最後に、複数のユーザプロフィールベクトルがNULLに初期化される。

【0154】（レギュラーシステム動作）正常動作中、インターネットユーザはウェブサイトを訪れ、ユーザの身分がログイン、またはウェブブラウザクッキーを介して確立される。ユーザ身分のそれぞれは、プロフィールベクトルに関連する。新しいユーザの場合、ヌルのプロ

フィールベクトルが作成され、プロフィールベクトルデータベースに追加される。ユーザがシステムに知られている場合、ユーザのプロフィールベクトルはプロフィールベクトルデータベースから検索される。

【0155】次いで、システムはユーザの行動を観察する。ユーザが探索照会217を行う場合、ユーザプロフィールベクトルは現在の照会のコンテンツベクトルを使用して更新される。ユーザが、218を読むためにウェブページを要求する場合、ユーザプロフィールベクトルは、要求されたウェブページのコンテンツベクトルによって修正される。

【0156】次に、更新されたプロフィールベクトルは、ユーザに表示するのに適した広告を選択するためのベースとして使用される。関連性は広告ベクトルデータベースに格納されているエンティティベクトルへのユーザプロフィールベクトルの近似性によって決定される。最も近いエンティティベクトルが、最も関連するものとして選択される。

【0157】ユーザに広告が提示され、その広告をクリックして広告の主題のさらなる情報を要求した場合、現在の広告エンティティベクトルおよびプロフィールベクトルの両方が更新される。ユーザプロフィールベクトルはエンティティベクトルの方向にわずかに移動もしくは適応する。エンティティベクトルは、広告をクリックしたユーザのプロフィールベクトルの方向にわずかに移動もしくは適応する。したがって、システムによって広告がリアルタイムに適応性のある方法でそれらを購入するユーザの特徴を想定することができる。

【0158】(発見および解析モード) 発見および解析モード中、システム管理者は、有用で利用可能なユーザ行動特徴を発見するため、エンティティベクトルデータベース230およびユーザプロフィールベクトルデータベース226の両方の内容を関係発見および解析を定期的な間隔で行う。RDAモジュール236は、ベクトルをクラスタ化し、ユーザのクリックスルー率の点でのシステムの有効性の統計情報の要約を報告し、提供する。解析中に発見された行動クラスタは、広告キャンペーンのベースとして機能し、ユーザ行動志向への価値のある洞察を行う。行動クラスタを、広告者が個人ベースよりもむしろグループ毎に対象としたいエンティティベクトルの初期状態として利用することができる。

【0159】図2に示すシステムは、適用可能な2つの構成要素のセットを有する。一つは、ユーザプロフィールベクトルである。二つ目は各広告のコンテンツベクトルである。両方のベクトルのセットは、ユーザの動作に基づいて互いに影響しあう。広告は、ユーザの興味が高い領域に向けて適応する方法で更新され、これによってシステム効率を向上させ、自動的にグループの志向を発見する。ユーザプロフィールベクトルはユーザの観察された動作に基づいて適応し、これによって個人およびグ

ループの両方のユーザ志向の正確で適時の表示を提供する。

【0160】ベクトルのセットはユーザの動作に依存し、時と共に発展するため、それらはユーザ集団の行動についての豊富な情報をコード化できる。これらのコンテンツベクトルのセットを、個人、グループの両方および集団全体を考慮した価値のある動作可能な洞察を提供するために、解析ツールを用いて「採掘」することができる。

【0161】上記の詳細な説明は、様々な実施形態に適応した本発明の基本的な新規の特徴を示し、記載し、指摘してきたが、本発明の意図から逸脱することなく、様々な省略、代替および変更が例示したシステムの形式と詳細について行われることは、当業者によって理解されるであろう。

【図面の簡単な説明】

【図1】本発明に関するコンピュータ環境を例示する機能的なブロック図である。

【図2】本発明の処理およびデータ構成要素を例示する機能的なブロック図である。

【図3】図2に示すワードコンテンツベクトル学習モジュールの前処理モジュールの入力および出力を例示するブロック図である。

【図4】図2に示すワードコンテンツベクトル学習モジュールの語幹ハッシュテーブルを作成するプロセスの入力および出力を例示するブロック図である。

【図5】図2に示すワードコンテンツベクトル学習モジュールの語幹生成プロセスの機能を例示するブロック図である。

【図6】図2に示すワードコンテンツベクトル学習モジュールの語幹頻度統計出プロセスを例示するブロック図である。

【図7】図2に示すワードコンテンツベクトル学習モジュールの異なる学習重疊ウィンドウの図である。

【図8】図7に示す重疊ウィンドウのための隣接する語幹のベクトルによって語幹ベクトルを適応するベクトル操作プロセスを例示する図である。

【図9】図2に示すセッションコンパイラのエンティティベクトル作成プロセスの入力および出力を例示するブロック図である。

【図10】図10aは、図2に示す高速ファイル作成プロセスの入力および出力を例示するブロック図である。図10bは、図2に示すセッション制御ファイルを生成するプロセスの入力および出力を例示するブロック図である。

【図11】図2に示すリアルタイム広告サーバモジュールが行う、ページ要求をコンテンツベクトルに変形するプロセスの入力と出力を例示するブロック図である。

【図12】図2に示すリアルタイム広告サーバモジュールによって行われる、照会をコンテンツベクトルに変形

するプロセスの入力と出力を例示するブロック図である。

【図13】図2に示すリアルタイム広告サーバによって行われる、広告選択プロセスの機能を例示するブロック図である。

【図14】図2に示すプロフィールベクトル更新サーバモジュールのプロフィールベクトル適応プロセスの機能を例示するブロック図である。

【図15】図2に示す関係、発見および解析モジュールによって行われるモデル構成プロセスの入力および出力を例示するブロック図である。

【図16】図2に示すプロフィールベクトル更新サーバモジュールが人口学的情報でプロフィールベクトルデータベースを更新するプロセスの機能を例示するブロック図である。

【図17】図2に示すエンティティベクトル更新サーバモジュールのエンティティベクトル適応プロセスの機能を例示するブロック図である。

【図18】図2に示すプロフィールベクトル更新サーバおよびエンティティベクトル更新サーバのベクトル適応プロセスを例示するフローチャートである。

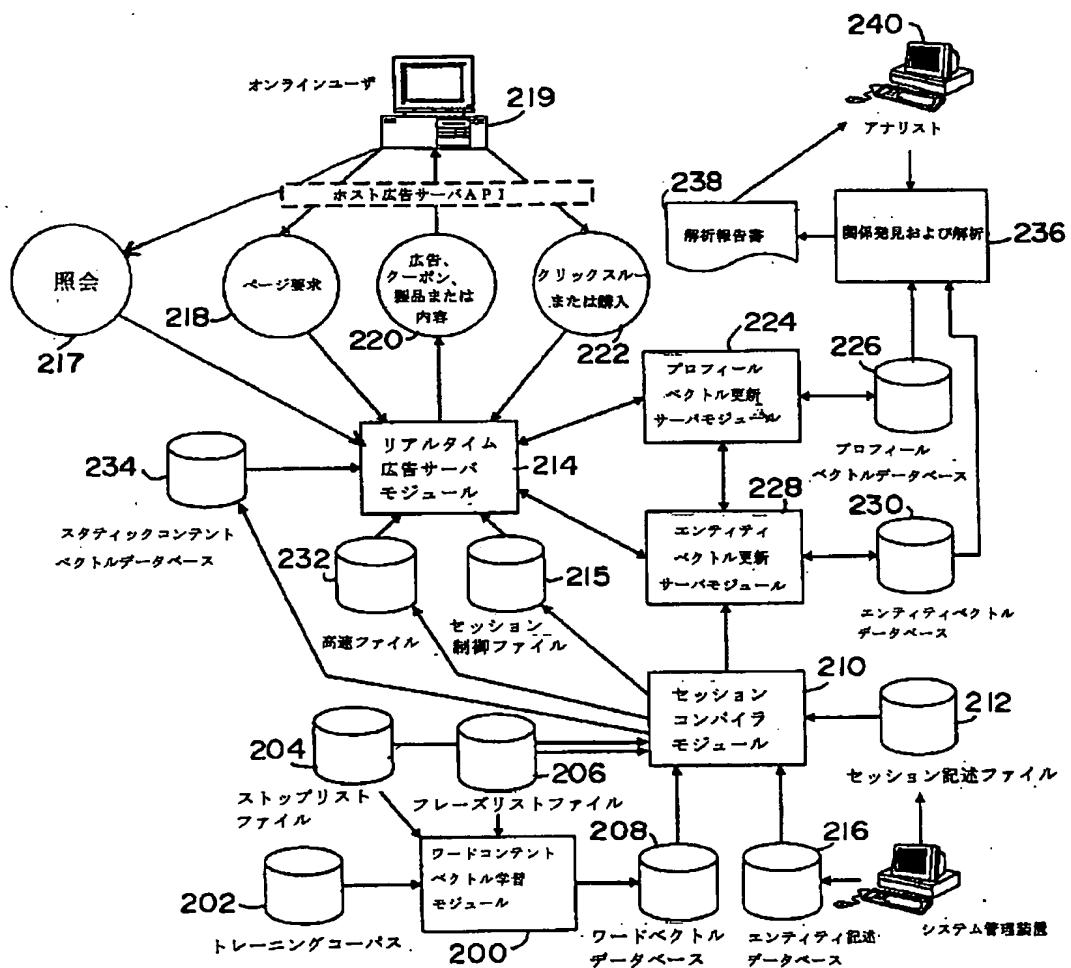
【図19】図2に示すプロフィールベクトル更新モジュールおよびエンティティベクトル更新モジュールによつて行われる固定率の学習アルゴリズムの数式流れ図である。

【図20】図19aおよび図19bに示す固定率の学習アルゴリズムのためのサンプルランタイムパラメータのサマリーを含むチャートである。

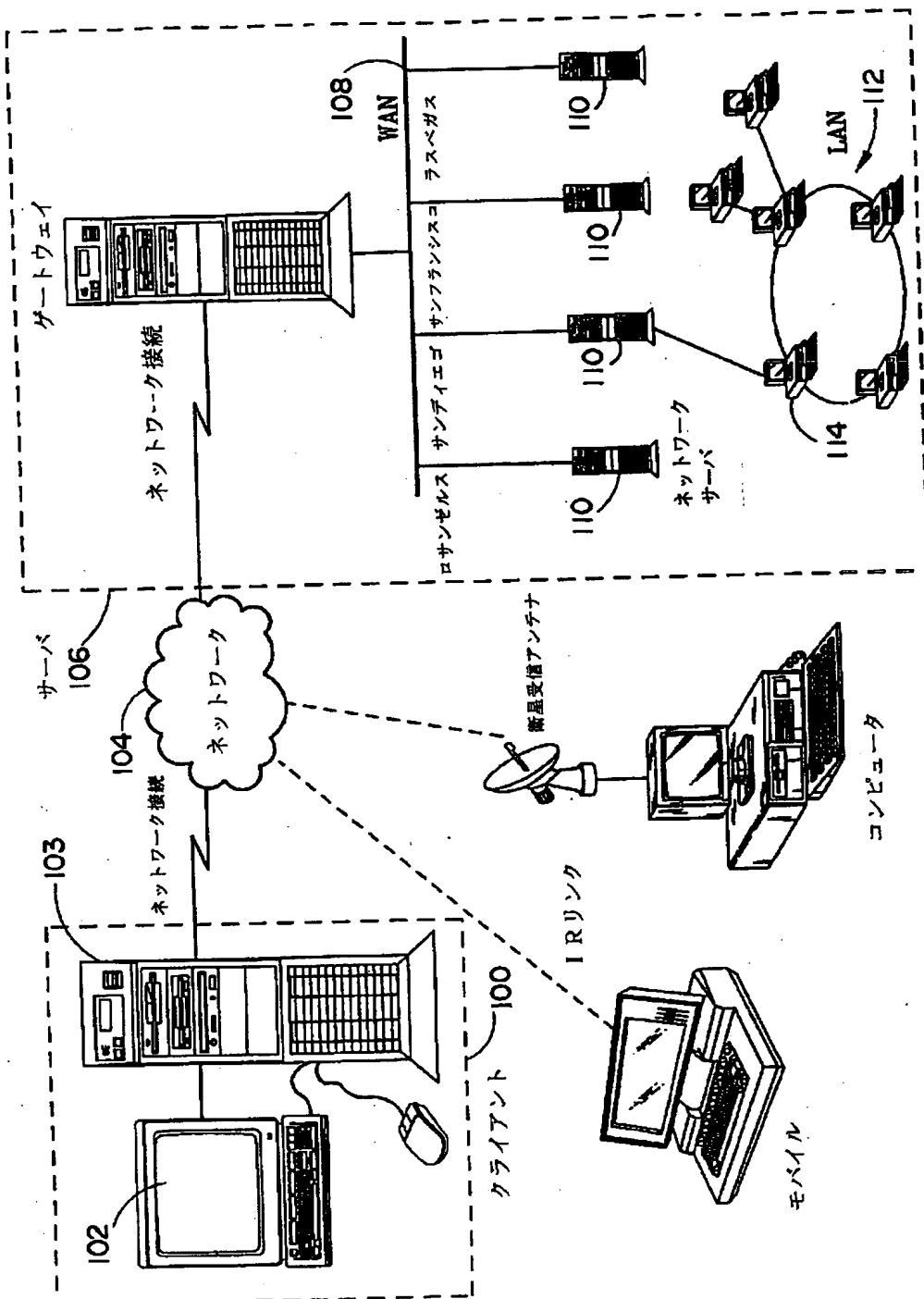
【符号の説明】

- 100 クライアント、
- 104 ネットワーク、
- 106 サーバ、
- 108 WAN、
- 110 ネットワークサーバ、
- 112 LAN

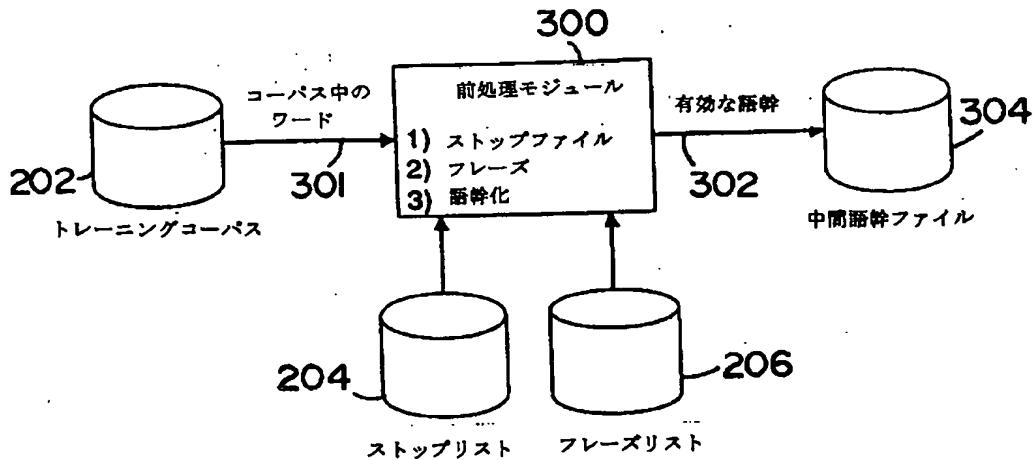
【図2】



【図1】

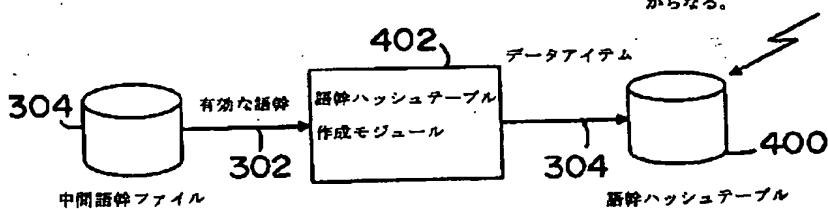


【図3】



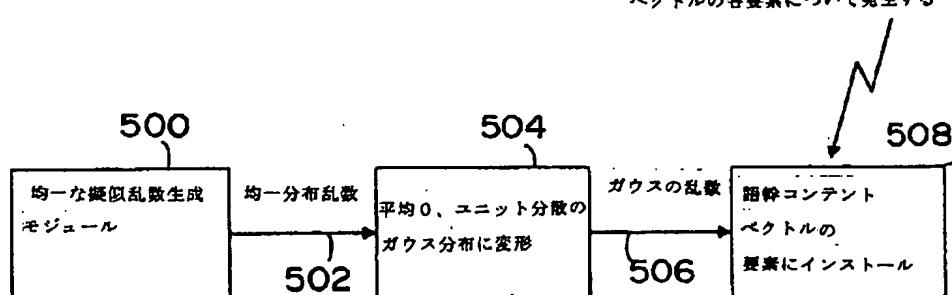
【図4】

語幹ハッシュテーブル中のデータアイテムは、
 - 語幹テキストストリング
 - 語幹コーパス統計
 - 語幹コンテンツベクトル
 - 積算された補正ベクトル
 からなる。

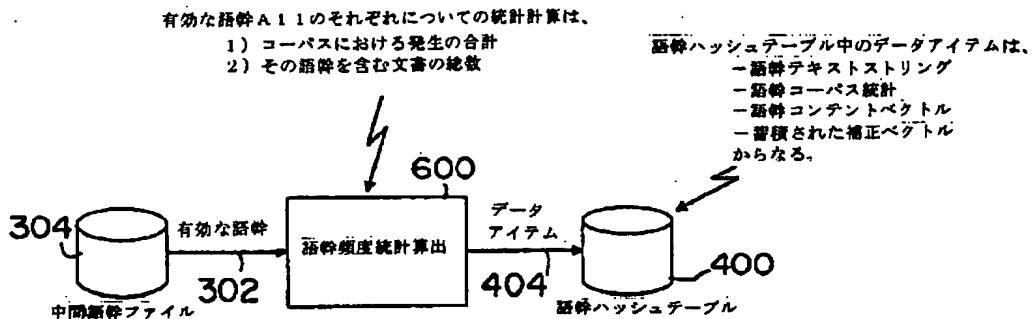


【図5】

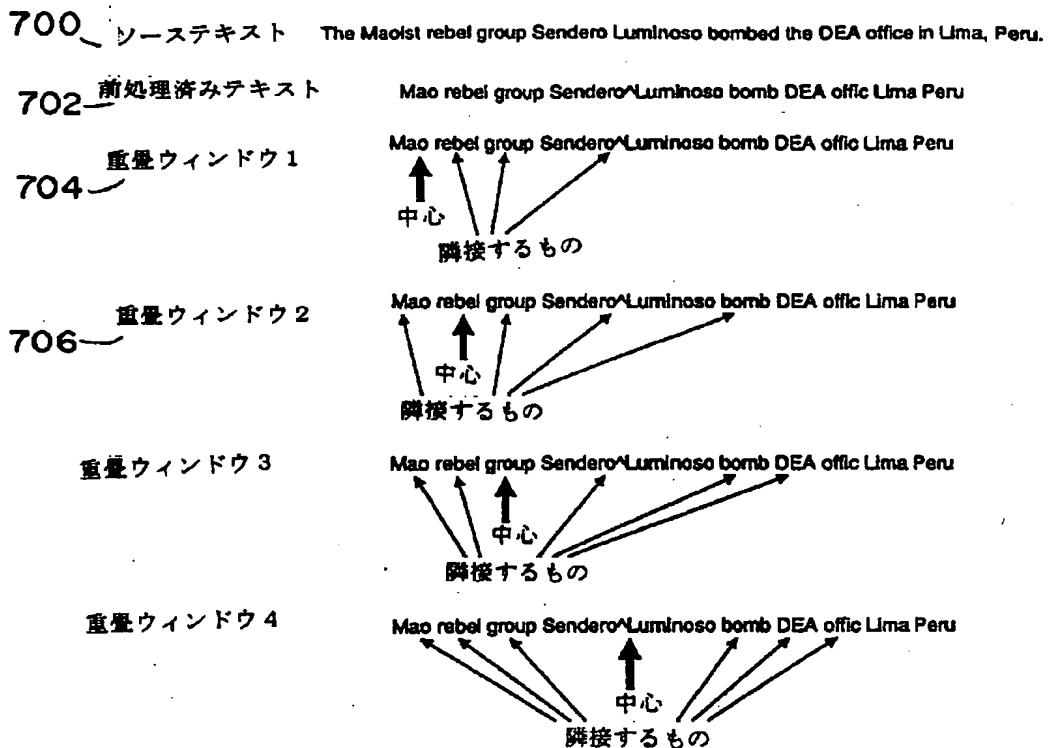
語幹コンテンツベクトルにおける乱数生成
 およびインストールは、各語幹コンテンツ
 ベクトルの各要素について発生する



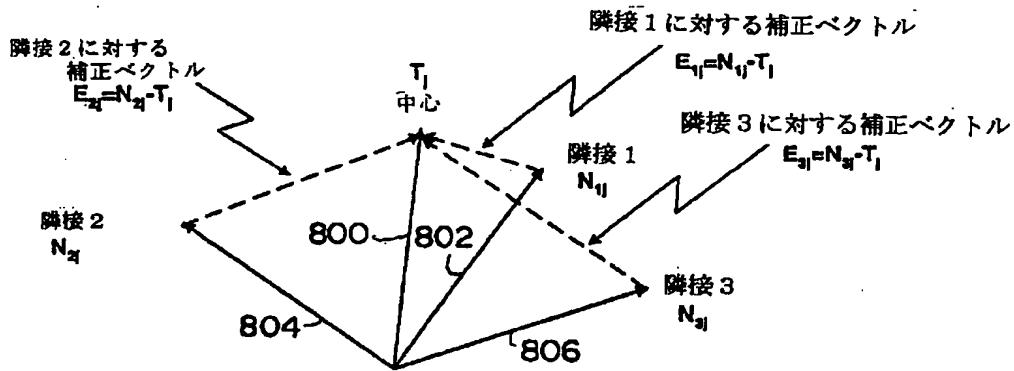
【図6】



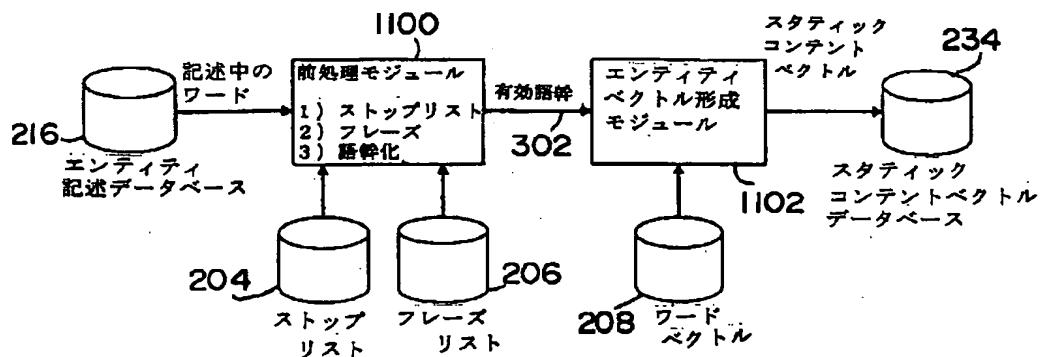
【図7】



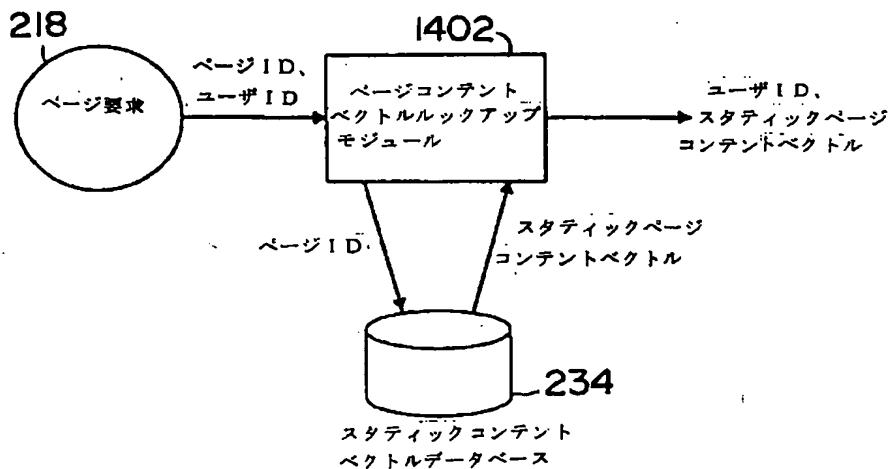
【図8】



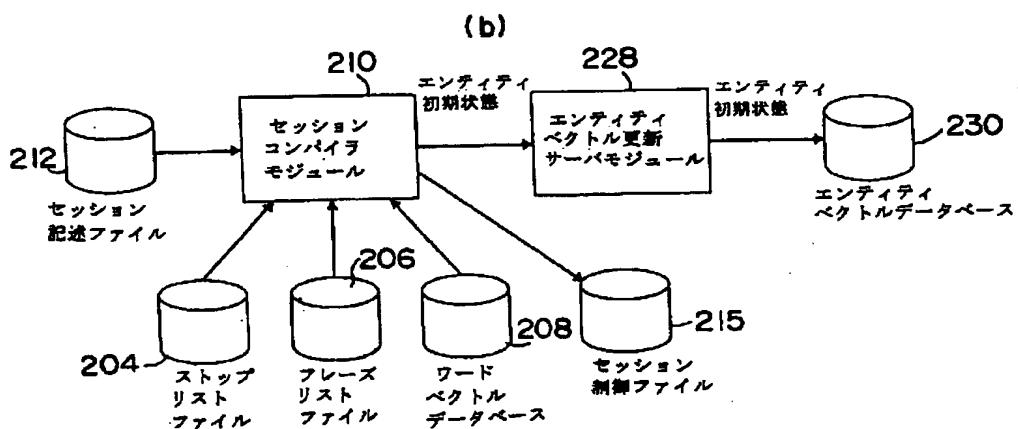
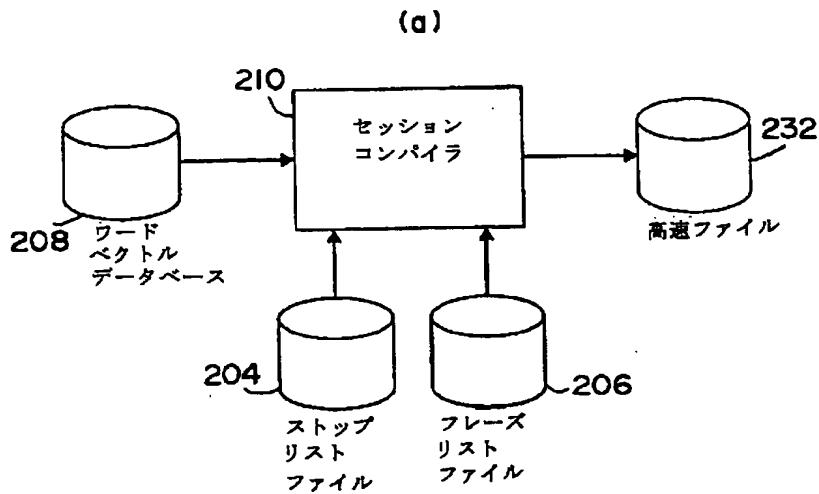
【図9】



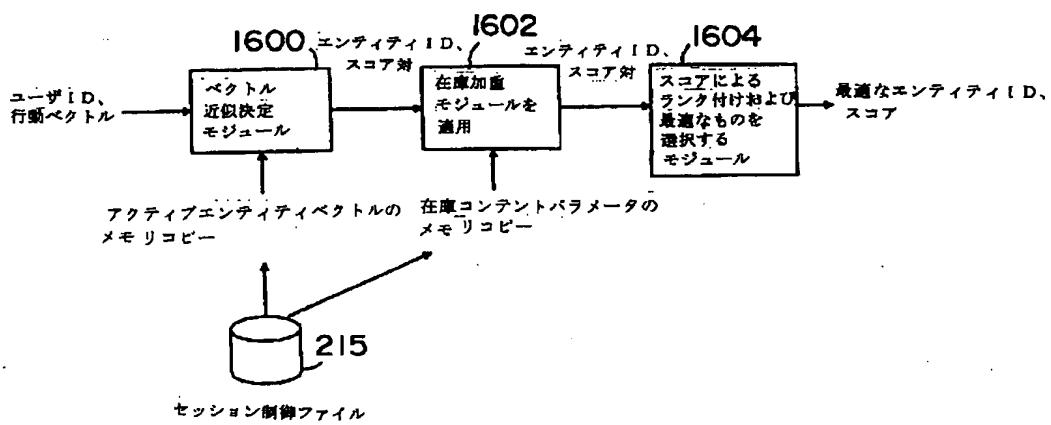
【図11】



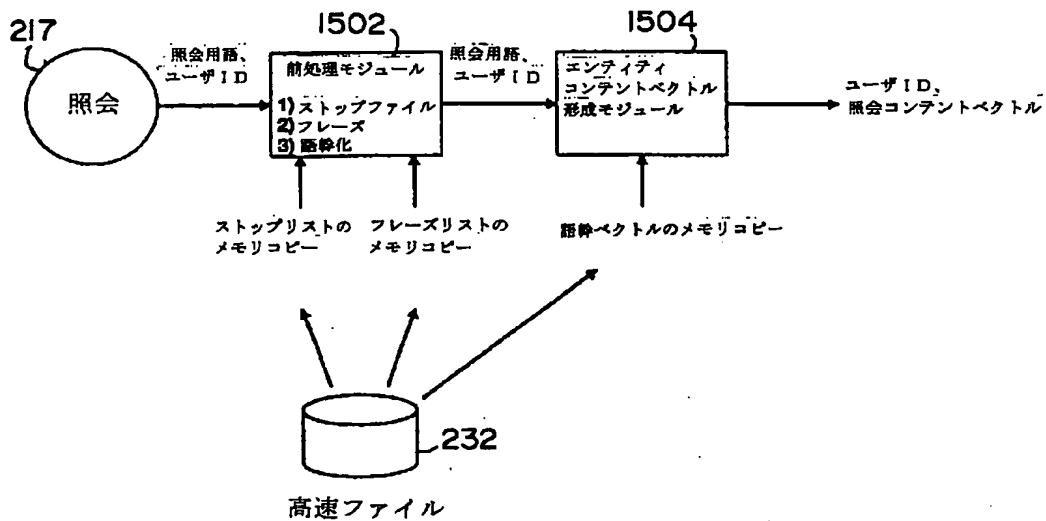
【図10】



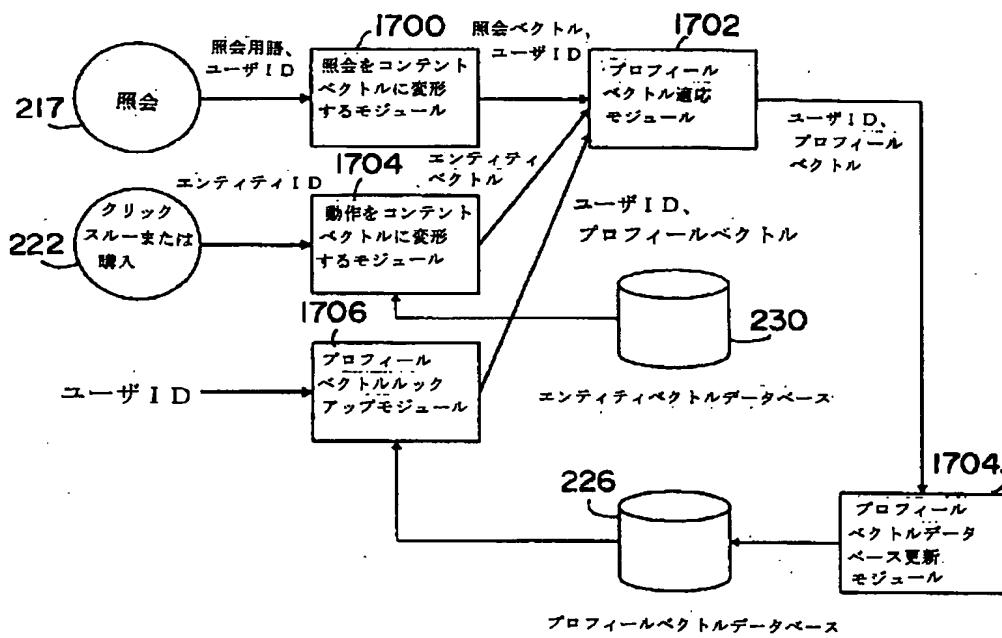
【図13】



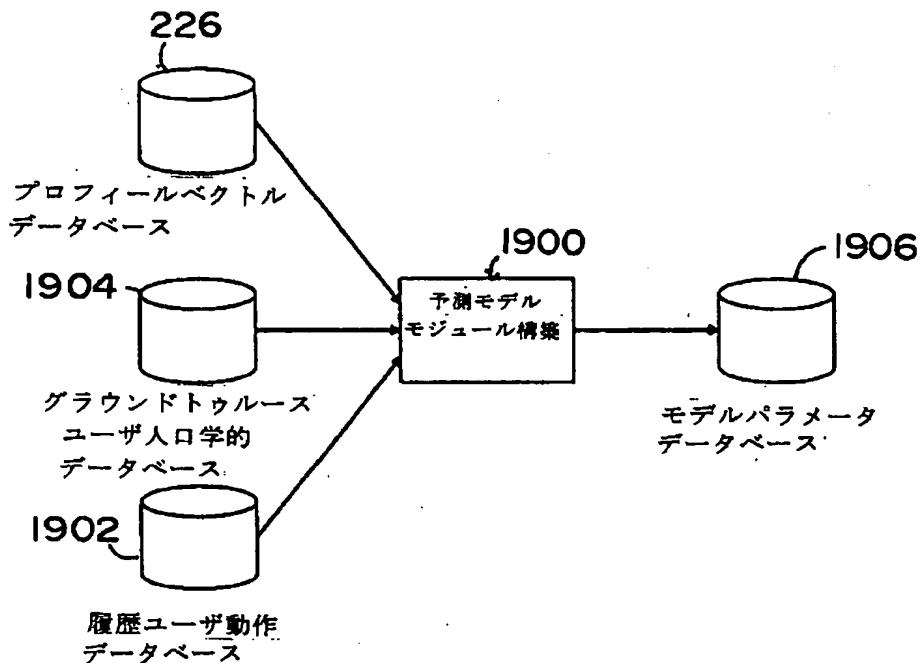
【図12】



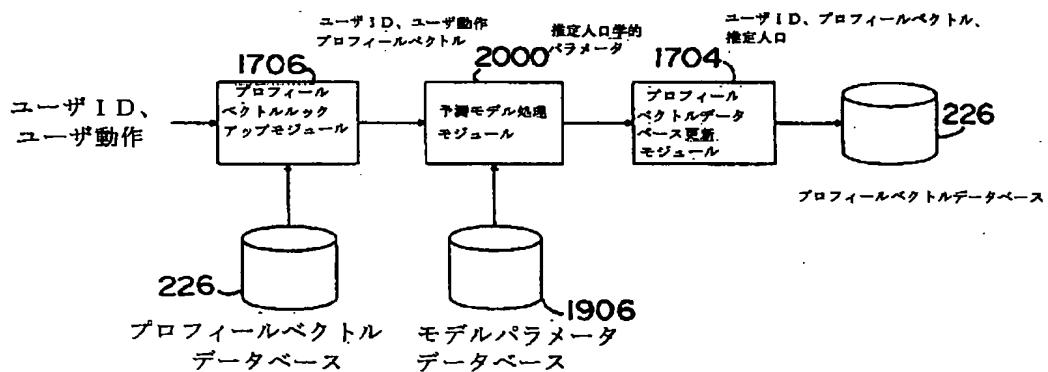
【図14】



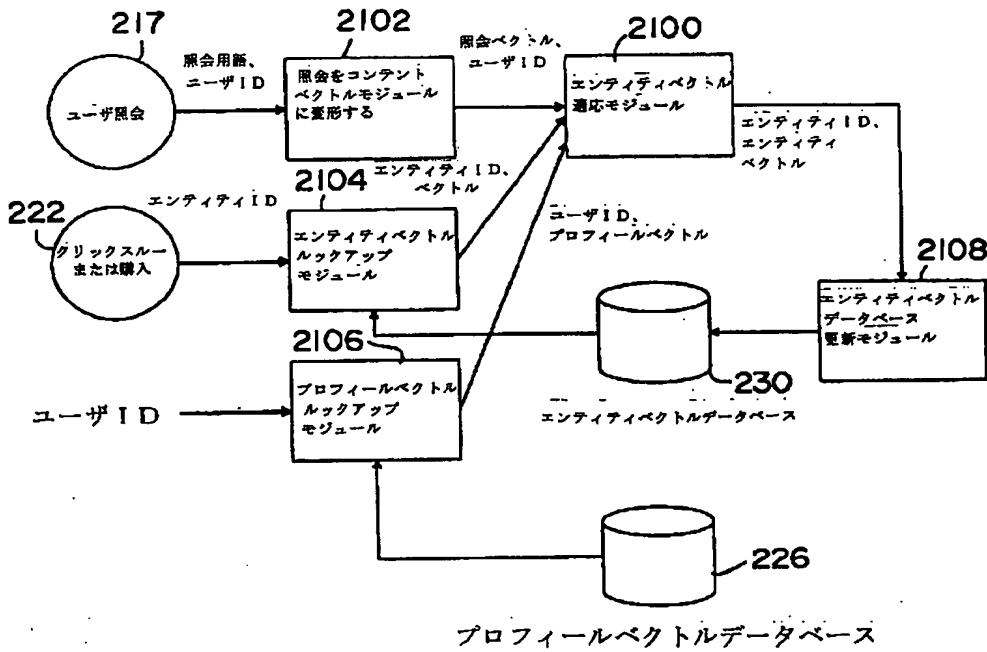
【図15】



【図16】



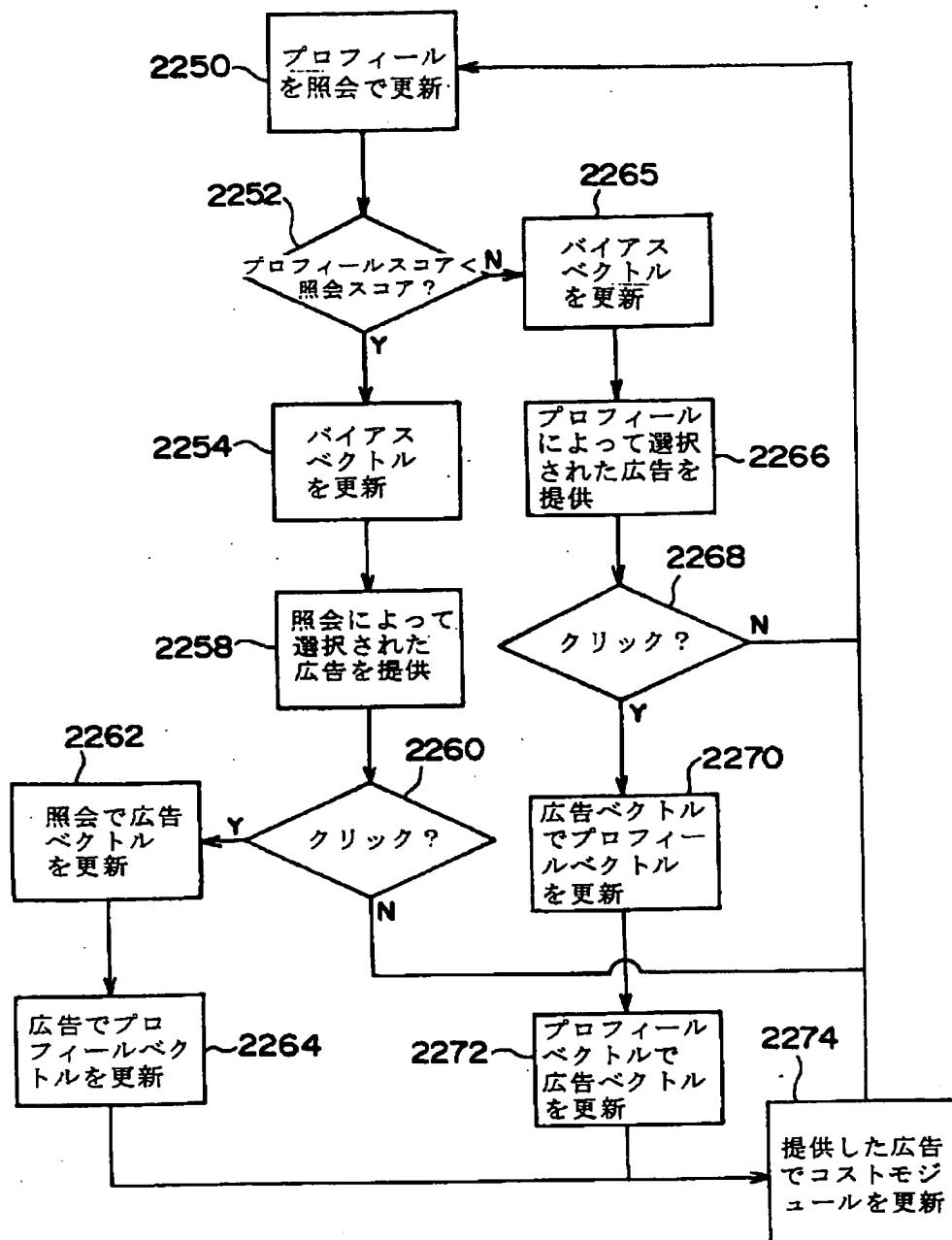
【図17】



【図20】

名称	典型値	値の範囲	説明
$threshold_{query}$	0.5	0.25-0.75	プロフィールベクトルの代わりに照会ベクトルを用いるしきい値
$\alpha_{profile}$	0.1	0.05-0.5	プロフィール更新についての学習率
α_{ad}	0.01	0.005-0.1	広告更新についての学習率
e_{query}	1.0e-06	1.0e-05 - 1.0e-07	照会ユニバースの平均の推定の更新率
$e_{profile}$	1.0e-06	1.0e-05 - 1.0e-07	プロフィールユニバースの平均の推定の更新率
e_{ad}	2.5e-05	1.0e-04 - 1.0e-06	広告ユニバースの平均の推定の更新率
$\Delta_{profile}$	0.01	0.001 - 0.5	プロフィールについての忘却因数
Δ_{ad}	0.01	0.001 - 0.5	広告についての忘却因数
μ_{ad}	0	アプリケーション依存	広告の平均
$\mu_{profile}$	0	アプリケーション依存	プロフィールの平均
μ_{query}	0	アプリケーション依存	照会の平均

【図18】



【図19】

(a)

```

if (q_score(Q_mf, A_1, ..., A_n) > p_score(P_mf, A_1, ..., A_n)) {
    [m, n, adddone] = arg max_{i,j,add} [q_i, p_j^{(add)} Value_{add} x ((q_i, p_j^{(add)}) > threshold_query)];
    r = arg max_j q_n * p_j;
    p_r = (1 - c_profile)p_r + c_profile(q_n - mu_query);
    p_r = p_r / |p_r|;

    if (bClickhere) {
        [k, l] = arg max_{i,j} [p_i, q_j^{(addone)}];
        s_n^{(addone)} = (1 - c_mf)s_n^{(addone)} + c_mf(q_n - mu_query);
        s_n^{(addone)} = s_n^{(addone)} / |s_n^{(addone)}|;
        p_k = (1 - c_profile)p_k + c_profile(s_n^{(addone)} - mu_mf);
        p_k = p_k / |p_k|;
        mu_mf = (1 - c_mf)mu_mf + c_mf s_n^{(addone)};
        mu_query = (1 - c_query)mu_query + c_query q_n;
        mu_profile = (1 - c_profile)mu_profile + c_profile p_r / |p_r|;

    } else {
        ;
    }
}

```

2276
2278
2280
2282

(b)

```

[m, n, adddone] = max_{i,j,add} [p_i, q_j^{(add)} Value_{add}];
q = arg max_j q_n * p_n;
p_m = (1 - c_profile)p_m + c_profile(q_n - mu_query);
p_m = p_m / |p_m|;

if (bClickhere) {
    s_n^{(addone)} = (1 - c_mf)s_n^{(addone)} + c_mf(p_m - mu_profile);
    s_n^{(addone)} = s_n^{(addone)} / |s_n^{(addone)}|;
    p_m = (1 - c_profile)p_m + c_profile(s_n^{(addone)} - mu_mf);
    p_m = p_m / |p_m|;
    mu_mf = (1 - c_mf)mu_mf + c_mf s_n^{(addone)};
    mu_query = (1 - c_query)mu_query + c_query q_n;
    mu_profile = (1 - c_profile)mu_profile + c_profile p_m;

} /* else */

```

2284
2286
2288
2290

フロントページの続き

(72)発明者 ウィリアム アール. ケイド
アメリカ合衆国 92117 カリフォルニア,
サンディエゴ ジェームスタウン ロード
5438番地

(72)発明者 リチャード エス. ブー
アメリカ合衆国 92064 カリフォルニア,
ボーウェイ ボーウェイ スプリングス
ロード 13639番地

(72)発明者 ブラッドレイ ディー. キンディグ
アメリカ合衆国 92064 カリフォルニア,
ボーウェイ ヘレン パーク レーン
14720番地

(72)発明者 ジェラルド エス. ラッセル
アメリカ合衆国 92130 カリフォルニア,
サンディエゴ カーメル ケイブ 12427
番地

(72)発明者 ケネス ピー. ブラウン
アメリカ合衆国 92126 カリフォルニア,
サンディエゴ ケンスリー ウェイ 6960
番地

(72)発明者 テッド イー. ダニング
アメリカ合衆国 92126 カリフォルニア,
サンディエゴ リバー リム ロード
11797番地

(72)発明者 ジョエル エル. カールトン
アメリカ合衆国 92129 カリフォルニア,
サンディエゴ アレンブルック ウェイ
8995番地

【外國語明細書】

1 Title of Invention

**SYSTEM AND METHOD FOR OPTIMAL ADAPTIVE MATCHING
OF USERS TO MOST RELEVANT ENTITY AND INFORMATION
IN REAL-TIME**

2 Claims

1 A computerized system for associating an observed behavior with informational items, comprising:

a converter capable of converting the observed behavior to a behavior vector;

a profile adapter capable of modifying a profile vector with the behavior vector; and

a comparater capable of comparing the modified profile vector to a plurality of entity vectors, the entity vectors indicative of the informational items, so as to identify at least one entity vector closely associated with the observed behavior.

2 A system for selecting advertisements in a computer environment, comprising:

a database of electronic advertisements; and

an electronic advertisement management system, comprising:

a converter capable of converting an observed behavior of a user computing device in the computer environment to a behavior vector,

a profile adapter capable of modifying a profile vector indicative of the user with the behavior vector,

a comparater capable of comparing the modified profile vector to a plurality of entity vectors, the entity vectors indicative of the electronic advertisements, so as to identify at least one entity vector closely associated with the observed behavior, and

a selector accessing the electronic database with the identified entity vector

P - 6 1 6 8 F

- 2 -

so as to select at least one electronic advertisement to communicate to the user computing device.

3 A computerized system for adapting an entity vector, comprising:

a converter capable of converting an observed behavior of a user into a behavior vector;

a profile adapter capable of modifying a profile vector indicative of the user based on the behavior vector; and

an entity adapter capable of modifying an entity vector indicative of an informational item based on the profile vector or the behavior vector.

4 A system for generating a profile vector in a computer environment, comprising:

a converter capable of converting a plurality of observed behaviors of a user into an associated plurality of behavior vectors; and

a profile adapter capable of repeatedly modifying a profile vector indicative of the user based on the plurality of behavior vectors.

5 A computerized system for adapting an entity vector, comprising:

a converter capable of converting an observed behavior of a user into a behavior vector; and

an entity adapter capable of modifying an entity vector indicative of an informational item based on the behavior vector.

6 A system for selecting advertisements in a computer environment, comprising:

a database of electronic advertisements; and

an electronic advertisement management system, comprising:

a converter capable of converting an observed behavior of a user computing device in the computer environment to a behavior vector,

a comparater capable of comparing the behavior vector to a plurality of entity vectors, the entity vectors indicative of the electronic advertisements, so as to identify at least one entity vector closely associated with the observed behavior,

P - 6 1 6 8 F

- 3 -

and

a selector accessing the electronic database with the identified entity vector so as to select at least one electronic advertisement to communicate to the user computing device.

7. A method of associating an observed behavior with informational items on a computer including a data storage, comprising:

converting an observed behavior to a behavior vector;

modifying a profile vector with the behavior vector; and

comparing the modified profile vector to a plurality of entity vectors, the entity vectors indicative of the informational items, so as to identify at least one entity vector closely associated with the observed behavior.

8. A method of selecting advertisements in a computer environment, comprising:

providing a database of electronic advertisements;

converting an observed behavior of a user computing device in the computer environment to a behavior vector;

modifying a profile vector indicative of the user with the behavior vector;

comparing the modified profile vector to a plurality of entity vectors, the entity vectors indicative of the electronic advertisements, so as to identify at least one entity vector closely associated with the observed behavior;

accessing the electronic database with the identified entity vector; and

selecting at least one electronic advertisement to communicate to the user computing device.

9. A method for adapting an entity vector on a computer including a data storage, comprising:

converting an observed behavior of a user into a behavior vector;

modifying a profile vector indicative of the user based on the behavior vector; and

modifying an entity vector indicative of an informational item based on the

P - 6 1 6 8 F

- 4 -

profile vector or the behavior vector.

10 A method of generating a profile vector in a computer environment on a computer including a data storage, comprising:

converting a plurality of observed behaviors of a user into an associated plurality of behavior vectors; and

repeatedly modifying a profile vector indicative of the user based on the plurality of behavior vectors.

11 A method of adapting an entity vector on a computer including a data storage, comprising:

converting an observed behavior of a user into a behavior vector; and

modifying an entity vector indicative of an informational item based on the behavior vector.

12 A method of selecting advertisements in a computer including a data storage, comprising:

providing a database of electronic advertisements;

converting an observed behavior of a user computing device in the computer to a behavior vector;

comparing the behavior vector to a plurality of entity vectors, the entity vectors indicative of the electronic advertisements, so as to identify at least one entity vector closely associated with the observed behavior;

accessing the electronic database with the identified entity vector; and

selecting at least one electronic advertisement to communicate to the user computing device.

3 Detailed Description of Invention

Background of the Invention

Field of Invention

The invention relates to associating entities and information with behavior. More particularly, the invention relates to a system and a process for targeting and delivering advertising, coupons, products, or informational content to users based upon

P - 6 1 6 8 F

- 5 -

observed behavior.

Description of the Related Technology

The widespread availability of the World Wide Web (web) and Internet services has resulted in a unique set of advertising opportunities. Unlike conventional "broadcast" media, such as television and radio, the web is a "narrowcast" medium that allows the user to have higher levels of control over the information they receive. Since users can control the retrieval of information, the advertising techniques utilized in the conventional broadcast model has become less effective and can alienate potential customers as a result of the "shotgun" effect. The potential of selectively targeting advertisements on a user-by-user basis has been unrealized due to the difficulty in performing meaningful targeting of customers. The current generation of web ad selection engines utilize a number of partially successful techniques to target customers. Typically, the effectiveness of these techniques is measured in terms of audience response rates. Audience response, also called "clickthrough", is evaluated by counting the number of users that click on a "banner advertisement" contained on a web page which is presented to the user. Clicking on the banner advertisement typically takes the user to the web site of the advertiser where additional information about a product or service is provided. In the current Internet advertising environment, clickthrough is the best measure of the effectiveness of advertising techniques. Consequently, the value of advertising is directly related to the effectiveness of the ad. Therefore, the maximization of clickthrough is of paramount importance for most web sites for both practical and financial reasons.

Some current banner advertising selection techniques are listed in the tables below. These techniques are divided into two classes. The first class of advertising selection techniques, shown in Table 1, use simplistic first generation techniques that are based only on static a priori information. The second class of advertising selection techniques, shown in Table 2, utilizes some of the techniques used in the more sophisticated second generation ad selection systems which may take into account some

measure of user behavior, such as a user query, or make use of predisclosed user preferences. However, these more sophisticated objectives often only complicate the problem. In Table 1 and Table 2 certain disadvantages of each method are given, however, the Tables are not meant to be all inclusive so not all disadvantages may be shown.

Table 1. First Generation Ad Targeting Techniques.

Method	Disadvantage
Domain name of user browser	<ul style="list-style-type: none"> • Provides no significant user insight; Does not account for user behavior • Implies user behavior is related to usage domain • Provides poor clickthrough
Browser type	<ul style="list-style-type: none"> • Implies user behavior is related to usage browser type/version • Provides poor clickthrough
System type	<ul style="list-style-type: none"> • Implies user behavior is related to system type • Provides poor clickthrough
Service provider	<ul style="list-style-type: none"> • Implies user behavior is related to usage domain • Provides poor clickthrough
Geography/location	<ul style="list-style-type: none"> • Requires up-to-date knowledge base of IP address versus location • Provides poor clickthrough • Provides no significant user insight
Site SIC Code	<ul style="list-style-type: none"> • Requires up-to-date knowledge base of SIC code versus IP address • Provides somewhat improved user clickthrough • Provides no significant user insight
Company size	<ul style="list-style-type: none"> • Requires up-to-date knowledge base of SIC code versus IP address • Provides no significant user insight • Implies user behavior is related to size of company

Knowledge base techniques	<ul style="list-style-type: none"> • Provides poor clickthrough • Requires maintenance of knowledge base • Costly to maintain • Cannot incorporate observed user behavior
---------------------------	---

Table 2. Second Generation Ad Targeting Techniques.

Method	Disadvantage
User registration information	<ul style="list-style-type: none"> • Requires user willingness to disclose information • Requires many judgments on products for effective operation • Requires a substantial preexisting judgment database
Search key words	<ul style="list-style-type: none"> • Misses ads that are conceptually close but do not contain key word • Restricts available inventory • Requires careful choice of key terms
Hand targeting of ads to sites or pages	<ul style="list-style-type: none"> • Very labor intensive; does not scale well • Humans not good at manual targeting for large numbers of ads
Hand analysis of site or locations in a site	<ul style="list-style-type: none"> • Very labor intensive; does not scale well • Humans not good at manual analysis of large numbers of sites
Random presentation	<ul style="list-style-type: none"> • Poor response rates • Alienates users
Date/time	<ul style="list-style-type: none"> • Poor response rates
Hand analyzed sites	<ul style="list-style-type: none"> • Very labor intensive • High cost

In the current web environment, users utilize search services to find relevant or interesting information. These search services provide a potential focus for the identification of user behavior as defined by the searches they execute, the web pages they view and positions in the directory hierarchy they visit. However, existing banner advertising selection techniques fail to analyze this behavior when selecting an advertisement. These search services provide an opportunity for presentation of user specific advertising.

The conventional techniques shown in Table 1 and Table 2, use hand coded associations and lists to perform customer "targeting." Certain techniques may use only the current user query as part of ad targeting. Each of the methods shown in Tables 1

P - 6 1 6 8 F

8

and 2 may be associated with five general categories of advertisement selection techniques. These categories include:

- (i) rule-based categorization;
- (ii) keyword based ad selection;
- (iii) assigning ads to individual web pages or sites;
- (iv) assigning ads to branches in hierarchical organizations of pages; and
- (v) collaborative filtering.

A description of each category is provided below.

Rule Based Ad Selection

Rule based ad selection uses available information and rules to select an appropriate ad. This technique can be effective if large numbers of rules are manually coded and the rule developer has a deep understanding of the problem domain. However, one problem with this system is that human intellectual effort is required to write or maintain the rules. The development of a rule base may be very expensive and time consuming.

Additionally, the rules for advertisement selection are limited to available variables with discrete values and provide "brittle" decision boundaries. These decision boundaries are typically binary and are frequently mishandled.

Rule based ad selection requires extensive knowledge about the targeted operating domain. Even with computer-aided tools, a knowledge engineer is required to develop the rules and administer the system. Furthermore, humans have demonstrated time and again, that they are poor at encoding rules. This observation is particularly true when large numbers of variables are encompassed within the scope of the problem being modeled.

Keyword Based Ad Selection

In keyword based ad selection systems, the ads are selected on the basis of one or more user provided words. When an observed user behavior (typically a user issued

P - 6 1 6 8 F

9

query) contains a known keyword, one of the ads, which is manually associated with the keyword, is selected for display. This technique provides good response rates for the keywords chosen. However, a major drawback with this approach is the system administrator must manually choose the keywords associated with each ad. This technique, based on intellectual effort and deep knowledge of the ad-specific domain, is time consuming and error prone. Additionally, the "inventory" of keywords at a site quickly becomes sold out. Lastly, with keyword based ad selection techniques, the ad selection process does not account for previous user behavior. The ad selector only uses a set of human-selected keywords in the current inventory based upon the current search query. Synonyms of user provided words are not automatically targeted without a thesaurus or synonym list.

Assigning Ads To Individual Pages

Ads are sometimes manually targeted to individual web pages. This method requires human intellectual effort to match an advertisement to a web page. However, such effort is usually prohibitive in large scale sites containing thousands of web pages.

Assigning Ads to Branches with Hierarchies of Pages

Ads are sometimes manually targeted to hierarchies or categorizations of pages called page ontologies. This technique may provide effective performance if the human making the assignment of the ad to a branch of the hierarchy has a good feel for the content of the site and understanding of the intended viewer. However, human intellectual effort is required to select the appropriate branch or category for the ad.

Assigning Ads To Web Sites

Ads are sometimes manually targeted to all of the pages in a web site. However, as was seen with many of the other systems, human intellectual effort is required to select the appropriate ad for the web site. This method is inflexible in that an advertiser cannot automatically target an ad to the best pages within a site.

P - 6 1 6 8 F

- 10 -

Typically, advertisers want to display different ads for each page in a site. Another problem with assigning ads to web sites is that the ads served to the user are not context dependent in that they fail to utilize user specific information.

Collaborative Filtering Techniques

Collaborative filtering (CF) techniques have been proposed for the problem of selectively targeting specific ads to web users. The CF approach requires large numbers of users to formally register with the system and make preference judgments about the quality of the ads, coupons and information content they receive. Under certain limited circumstances, this technique may provide some degree of effective matching in small scale tests. However, for large tasks, such as found on the Internet, the algorithms become both computationally intractable and impractical. Additionally, the CF techniques perform best when the universe of entities are static rather than time evolving. In the Internet market, static content is the exception, not the norm. Furthermore, the utility of the CF approach is directly related to the numbers of users that have made preference judgments. Typically, users are reticent to spend a substantial amount of effort which is required to make these judgments both because of personal time limitations and concerns about privacy. As such, data acquisition becomes a major problem for the advertiser.

CF solutions typically require a knowledge engineer to implement a system of sparse vectors of a very high dimension, where the dimension of the problem space is equal to the number of entities under consideration. However, this solution has been found to be computationally intractable. Another problem with CF techniques is that the ad selection software is examining for orthogonal relationships which requires a substantial number of statistically complicated steps. The ad selection software must find close neighbors in the vector space and suggesting coordinates not in common between the neighbors.

The existing ad selection systems, particularly in coupon and print media

P - 6 1 6 8 F

- 11 -

advertising, fail to address the interactive nature of the Internet and electronic commerce. Advertisers need to be able to identify users of specific interests, track those interests over time and disseminate, in a highly selective way, information, advertising, coupons and product offerings that will be of interest to the user. Additionally, advertisers need to track user interests and behavior in a real-time manner.

Therefore, advertisers need a system which is sensitive to user behavior for targeting advertising, coupons, products and information content. This system should enable the targeted marketing in real-time and to a granularity of an individual user as well as groups of users that have similar behavioral characteristics.

Summary of the Invention

The present invention includes a system and method for delivering targeted entities, such as advertising, coupons, products and information content, to users based on tracking observed behavior. The system model is an adaptive vector space representation for entities, information and behavior. The system matches users to entities in a manner that improves with increased operation and observation of user behavior.

The present invention uses an informational representation called entity or content vectors. The content vectors are a high dimensional, real vector representation that encodes contextual similarity. The system utilizes a constrained self-organization learning algorithm to learn the relationships between symbols which are typically words in unstructured text. This learning algorithm discovers relationships between words or symbols from a set of training examples for a domain of interest and assigns a unique content vector to each symbol. The resulting representation is such that symbols that appear in a similar context will possess content vectors that point in similar directions in this high dimensional content vector space.

Web pages, banner advertisements, coupons, and any other textual or symbolic information, are represented as a summary content vector by forming the normalized weighted vector sum of the content vectors of the words (symbols) in the page or

P - 6 1 6 8 F

- 12 -

description of the product or service. These summary content vectors have the property that pages that discuss similar information have summary content vectors that point in similar directions.

The actions of each user are represented by profile vectors. Typical user actions include, but are not limited to: issuing queries requesting or reading pages, visiting Internet sites, responding to ads, and redeeming coupons and purchasing products. A user's profile vector is configured to track observed behavior by using the content vector representation of the associated actions. For example, when the user requests a page of information, the user's profile vector is adjusted based upon the entity vector of the requested page.

The user profile vector is utilized as the mechanism for selecting the best entity (e.g., ad, coupon, product or information) to present to the user. The user profile vector is compared to each active entity vector, and the closest entity content vector is selected for display to the user.

The present system also performs the optimal adaptive positioning of entity vectors. If a user is presented a banner ad and requests more information by clicking the mouse on it, the entity vector for the ad is adjusted using the user's profile vector. Additionally, the user profile vector is adjusted using the ad content vector. This mutual adaptation allows optimum positioning of ads relative to users.

In this system, the user profile vectors evolve through a user's usage of the system. Clustering and classification techniques are applied to the adapted profile vector set to discover behavior patterns that are utilized for focused marketing campaigns.

Likewise, ad and coupon entity vectors evolve as a result of user behavior. These evolved vectors encode summary characteristics of the users who clicked on the ad or redeemed the coupon. These summary characteristics are used to modify product positioning and discover the characteristics of the users who are interested in the product or service.

The system is fully automated and does not require any rules for operation and is

P - 6 1 6 8 F

- 13 -

based on self organizing neural network techniques. Additionally, the system operates using tacit user observation and does not require explicit user judgments for correct operation.

Detailed Description of the Preferred Embodiments

The following presents a detailed description of certain specific embodiments of the present invention. However, the invention can be embodied in a multitude of different ways as defined and covered by the claims. In this description, reference is made to the drawings wherein like parts are designated with like numerals throughout.

Figure 1 is a block diagram illustrating a computer environment associated with the present invention. A client computer 100 has a monitor 102 and a processing unit 103. The processing unit 103 includes a data storage.

The client computer 100 stores information that may be exported to other computing devices through a network 104. The network 104 may include any type of electronically connected group of computers including, for instance, the following networks: Internet, Intranet, Local Area Networks (LAN) or Wide Area Networks (WAN). In addition, the connectivity to the network may be, for example, remote modem, Ethernet (IEEE 802.3), Token Ring (IEEE 802.5), Fiber Distributed Datalink Interface (FDDI) or Asynchronous Transfer Mode (ATM). Note that computing devices may be desktop, server, portable, hand-held, set-top, or any other desired type of configuration.

When a remote user requests information from the network 104, the client computer 100 displays an advertisement with the requested information. Based on a request from the client computer 100, a server 106 may select an appropriate advertisement to display to the user based upon the user's observed behavior.

In one implementation, the server 106 includes a gateway which is connected to a WAN 108. The WAN 108 has a plurality of network servers 110. One of the network servers 110 is connected to a LAN 112 comprising a plurality of computers 114. The system software that selects the advertisement may be located on one the network servers 110 or another computer in the network 104. In one embodiment of the invention, the

P - 6 1 6 8 F

- 14 -

advertisement selection software executes in part on a plurality of the network servers 110. In another embodiment of the invention, the advertisement selection software executes on a plurality of the computers 114 on the LAN 112. In yet another embodiment of the invention, the advertisement selection software resides on the client computer 100. It is important to understand that the advertisement selection system of the present invention may be hosted on any computing device so long as a communication pathway exists between the advertisement selection software and the requester of the advertisement.

Figure 2 is a functional block diagram illustrating the processing and data components of the present invention. The following description focuses on the system's ability to deliver advertising, specifically banner advertising over the Internet. However, the techniques utilized in this invention are general in nature and can be applied to any type of entity or information. Furthermore, the network environment need not be the Internet as indicated previously.

The present invention employs vector modeling of information so as to identify relationships among the informational items or entity vectors. A process for making and using entity vectors is described in the U.S. Patent No. 5,619,709 filed November 21, 1995, entitled "SYSTEM AND METHOD OF CONTEXT VECTOR GENERATION AND RETRIEVAL" to Caid, et al.

This entity vector generation scheme is designed to produce content vectors. In an application to words, for instance, the scheme generates one vector per word, which represents the relative proximity of a meaning or content among words or documents in a quantitative and geometric manner. The system then creates an entity vector based upon the words in the entity and their associated word content vectors. Thus informational items having similar meanings have closely aligned vectors, while informational items having dissimilar meanings have approximately orthogonal vectors.

In the following description, the words "content vectors" and "entity vectors" are used synonomously. Both terms refer to a vector that is associated with certain content information, ad, coupon, solicitation, or product, for instance.

The following description is divided into the sections: (1) System Component

Overview, (2) Word Content Vector Learning Module, (3) Session Compiler Module, (4) Real-Time Ad Server Module, (5) Profile Vector Update Server Module, (6) Entity Vector Update Server Module, (7) Profile and Entity Adaptation Algorithms, (8) Calculation of Entity Inventory Weighting (Cost Model), and (9) Relationship Discovery and Analysis Processing Module.

System Component Overview

Beginning the discussion by reference to the bottom of Figure 2, a word content vector learning module 200 creates a set of word content vectors from a domain relevant training corpus 202, a stop list file 204, and a phrase list file 206. The stop list file 204 contains a list of words, word fragments and phrases for the word content vector learning module 200 to ignore. The phrase list file 206 is a list of words that the word content vector learning module 200 groups together. The training corpus 202 is the set of informational objects used to train the word vectors. The functions of the training corpus 202, the stop list file 204, and the phrase list file 206 are further described below.

When the word content learning process is complete, there is one word content vector in a word vector database 208 for each word in the training corpus 202 that is not in the stop list 204. These word content vectors have the property that words that are used in a similar context have content vectors that point in similar directions in the high dimensional content vector space. The word content vectors serve as the basis for the ad selection operations performed by the present system during its operation. The word content vector learning process is performed once as part of the initialization process. However, these vectors can be updated if desired. Normally, the word content vectors are static during normal operation of the invention, since the meaning of words in the English language evolves slowly over time.

A session compiler module 210 takes a set of session control information, in the form of structured text from a session description database 212, and transforms the text into control commands for a real-time ad server (RTAS) module 214 in a session control file 215. Typical information processed during the compilation process includes

the number of active ads, the number of allowable impressions for each ad, start and stop times, and dates for showing each ad. The session compiler module 212 may be omitted from the system if such functionality is not needed.

The present invention then computes an initial entity vector for each ad or entity in the system. In most cases, this entity vector is based on a textual description of the entity which is located in an entity description database 216. For instance, if the entity is an ad for a car, the words in the advertisement are used to determine the entity vector. The entity vector is computed as a normalized weighted sum of the word content vectors that are in the advertisement. A textual description of any item can be used as the initial condition for the entity vector. Since entity vectors can evolve during system operation, the initial condition of the entity vector can be null, and no textual description of the entity is required for system operation. However, in normal operation, this evolution may require substantial time if system usage rate is low and as a consequence, the preferred embodiment would start with any reasonable text description as an initial condition.

In the present invention, more ads can be sold in a thematic region of interest than in previous systems since all ads are selected on the basis of vector comparisons and not exact matches, thus increasing the available inventory. For example, the provider of the present invention could sell an ad slot based upon the regions of the vector hyperspace close to the words in the ad and not just a selected set of words.

The entity vector approach also allows administrators to effectively setup ad definitions without special skills or domain knowledge, thus improving overall system performance without the requirement for domain experts. The present invention automatically targets an ad to the best pages within a site, not just the site as a whole. As a consequence of this feature, different ads may be associated with the same page at a site. In addition, the present invention automatically selects the most relevant pages within a hierarchy or category for a particular ad. Alternatively, the system can be configured to serve the best ad independent of any ontology that might be present and based upon only on matching to observed user behavior.

P - 6 1 6 8 F

17

In the present invention, the ads served to the user are context dependent in that they depend on both the contents of the page, the current user action and the historical user profile. The entity vector approach provides a solution which is compact and adaptive. The entity vector representation requires a much smaller amount of data for training as compared to the collaborative filtering approaches. Additionally, the multidimensional entity vector representation of the present system leads to superior generalization capabilities as compared to the collaborative filtering approaches.

Referring again to Figure 2, the real-time ad server (RTAS) module 214 is the engine which observes the user behavior, and based upon that behavior, selects the appropriate ad to present to the user computer 219 located at the top of the figure. The RTAS module 214 uses the current user action, the user's profile vector and the active set of entity vectors in selecting the appropriate advertisement. The RTAS module 214 uses vector proximity calculations such as inner product to determine the optimal ad to display to the user. However, other distance metrics may also be employed.

If the RTAS module 214 detects that a user has performed an "adaptation event" on the client computer 219 such as issuing a query 217, requesting a page 218, purchasing a product or clicking on an ad 222, the RTAS module 214 sends an update message to the a profile update server (PVUS) module 224 or an entity update server (EVUS) module 228. The RTAS module 214 utilizes a static content vector database 234 and a high speed file 232 to hold working values of the entity or ad vectors. Upon the selection of the proper entity, the RTAS module 214 sends the selected ad to the client computer 219.

The PVUS module 224 adapts the user profile vectors in response to a request from the RTAS module 214. A profile vector for each user is stored in a profile vector database 226. This adaptation process moves the user profile vector in the direction of the entity vector associated with the adaptation event. For instance, if the user clicks on an ad, the user profile vector is moved a small step in the direction of the entity vector for the ad.

The EVUS module 228 performs a nearly identical action as the PVUS module

P - 6 1 6 8 F

- 18 -

224, but performs the operation on the entity vectors for advertisements. Each entity vector is stored in an entity vector database 230. For example, if the user clicks on an ad, the entity vector for the user selected ad is moved a small step in the direction of the user's profile vector.

A relationship discovery and analysis (RDA) module 236 performs processing actions on the contents of both the profile vector database 226 and the entity vector database 230 to identify useful patterns and trends. The RDA module 236 utilizes both clustering of the entity vectors and profile vectors to identify useful behavior patterns.

It should be noted that the adaptation of the entity vectors is not required for system operation. Indeed, when adaptation is enabled, it allows the entity vector of the product to "drift" toward users who have received the ad. In some applications, information content delivery systems in particular, this drift is not desirable. Clearly, the information content of a news story is deterministic and should not change based on who reads the story. As such, the present invention has provisions to disable the adaptation of entity vectors. However, in many applications, such as Internet advertising, the drift of the vectors is the mechanism whereby the system discovers patterns and trends.

In the preferred embodiment described herein, the software programs are written in the C, C++ and TCL programming language and run under the well-known UNIX operating system. C, C++, and TCL are industry standard programming languages for which many commercial compilers can be used to create executable code. However, the present invention may be run on a variety of conventional hardware and software platforms. The software system described herein may be distributed across different subsystems running on different hardware platforms. The components may communicate using messages transmitted using any network protocol, such as Transmission Control Protocol/Internet Protocol (TCP/IP) and RCP. The specific platform that hosts these modules and files is unimportant to the invention provided that the host computing environment has sufficient processing power. Additionally, each processing and/or data component may be hosted on a separate platform as long as paths

of connectivity are provided via any sufficiently robust communication protocol.

The present invention includes six principle processing modules as follows: the word content vector learning module 200, the session compiler module 210, the real-time ad server module 214, the entity vector update server module 228, the profile vector update server module 224, and the relationship discovery and analysis module 236. Below, the aforementioned components are described in further detail.

Word Content Vector Learning Module

The word content vector learning module 200 (shown in Figure 2) is implemented as an application programming interface. The application programming interface may be part of a computer program library. Advertisers use the procedures provided by this library to create a stand alone executable computer program capable of running on any platform of sufficient processing power. This program performs the content vector learning process which in one embodiment, is further described in the Caid patent. Additionally, this program writes the output of entity vector learning into data structures which are then utilized by the session compiler module 210 to create files optimized for runtime efficiency.

The training corpus 202 which includes a large number of documents containing unstructured text or symbols is selected and processed through the content vector learning algorithm. The resulting objects produced through the content vector learning process are then adapted for use by the other system components.

Figure 3 is a block diagram illustrating the inputs and outputs of a pre-processing module 300 of the word content learning module 200 shown in Figure 2. As part of the pre-processing the word content vector learning performs the following functions:

1. Forms input.

Referring to Figure 3, the requisite input files for the word content vector learning module 200 are assembled. The three input files are the training corpus 202, the stop list file 204 and the phrase list file 206.

P - 6 1 6 8 F

20

In the situation where the symbols of interest are a series of words 301, the training corpus 202 includes a set of unstructured free form text documents that cover the domain of interest. For example, if the domain of interest is outdoor sports such as hunting or fishing, an appropriate training corpus might be 3 years of articles from Field and Stream magazine and the outdoor section of the local newspaper. This text is typically a "flat" ASCII format with any typesetting or formatting information removed.

The stop list file 204 (shown in Figure 2 and 3) is developed through a combination of human intellectual effort and conventional statistical techniques, that contains a list of words, word fragments and phrases for the word content vector learning module 200 to ignore. For the content vector word learning, as with conventional statistical information retrieval techniques, many words are meaningless. For instance, "AND", "NOT", "BUT", and "THE" are words that convey little meaning in terms of determining the thematic content of a set of text. As a consequence, these terms should be ignored. The stop list file 204 is a flat ASCII file that contains an exhaustive and enumerated list of words to ignore with one entry per line. The typical length of this file for conventional applications is approximately 250-350 of the statistically most frequently occurring words in the training corpus 202.

The phrase list file 206 (shown in Figures 2 and 3) contains the list of word sequences that are to be treated as a single symbol. Using this file, the word content vector learning module 200 treats certain phrases as single symbols. For instance, "San Diego" should be considered as a single symbol. The contents of the phrase list 206 are determined through intellectual effort, statistical techniques, linguistic processing or any combination thereof. The phrase list file 206 is not required for correct operation of the word content vector learning module 200, but the use of a meaningful phrase list does improve system performance. The phrase list 206 is a flat ASCII file with one phrase per line in the file.

2. Performs word content vector learning.

Once the training corpus 202, stop list file 204 and phase list file 206 have been assembled, the word content vector learning process is performed. The Caid patent

P - 6 1 6 8 F

- 21 -

provides a full disclosure on this approach, but a summary of the key steps are repeated here for convenience.

Referring to Figure 3, the preprocessing is performed by a preprocessing module 300 which is part of the word content vector learning module 200. The preprocessing module 300 reads the training corpus 202, determines if the current word is on the phrase list file 206 or the stop list file 204, performs the stemming of the word, and writes a resulting valid stem 302 to an intermediate stem file 304 (shown in Figure 3). The stemming process removes common suffixes from each of the words in the training corpus 202.

Figure 4 is a block diagram illustrating the inputs and outputs of the process for creating a stem hash table 400. The stem hash table 400 is created by a stem hash table creation module 402. The stem hash table creation module 402 is a sub-component of the word content vector learning module 200. For all of the stems 302 in the intermediate file 304, data items 404 are made in a stem hash table 400. This hash table also contains pointers for each stem to a unique data element that consists of a stem text string, stem statistics, a stem content vector, and a correction content vector that is used as temporary storage to accumulate the change vector to be applied to the stem content vector as part of the learning process.

Figure 5 is a block diagram illustrating the random number generating process for creating initial conditions for the stem vectors. The initial conditions for each stem data item in the stem hash table 400 is created by a uniform pseudorandom number generator (UPNG) module 500. The UPNG module 500 generates an initial random content vector and stores this vector in the stem content vector. The UPNG module 500 generates initial conditions for stem vectors through the use of a long period pseudorandom number generator which creates a flat distribution random number 502. Any random number generator can be used assuming the period of repetition is sufficiently long (more than 100,000,000 cycles). Using the flat distribution random number 502, a transformer module 504, creates a Gaussian random variate 506 that has the property of zero mean and unit variance. An installation module 508 assigns the

P - 6 1 6 8 F

- 22 -

first Gaussian random variate to the first element of the current stem content vector. A second random variate is then assigned to the second element of the stem content vector. This process of assigning unique Gaussian distributed random numbers is repeated until all elements of all stem content vectors of all the stems in the stem hash table 400 have been populated.

Figure 6 is a block diagram illustrating the stem frequency statistics calculation process of the content vector learning module shown in Figure 2. A processing module 600 acquires the stem frequency statistics from the intermediate stem file 304. The processing module 600 is a sub-component of the word content vector learning module 200. The processing module 600 reads the intermediate stem file 304 and computes the number of occurrences of each stem 302 in each document in the intermediate stem file 304. The processing module 600 then stores these statistics in the stem statistic section of each data item 404 in the stem hash table 400.

Figure 7 is a diagram of the different learning convolutional windows of the entity selection system shown in Figure 2. A series of corrections to the initial conditions of the stem content vectors based on word co-occurrences are computed for each of the stem content vectors in the stem hash table 400. The present system modifies the stem content vectors based on the words that co-occur with a stem using a convolutional co-occurrence window. Figure 7 shows an initial source text 700. After some preprocessing, the source text 700 is transformed into the preprocessed text 702. The system creates a convolutional window 704 to analyze a center stem and the center stem's neighbors. The window has a center and up to three words on either side of the center, thus providing seven total words in the window. As can be seen in Figure 7, the center of the window 704 is at the front of the document. As such, the window is one-sided. The correction vector for the window 704, centered on the stem "mao", consists of a vector function of the initial condition content vectors for "rebel", "group", and the phrase "Sendero Luminoso." The correction vector for a window 706, centered on stem "rebel", consists of a vector function of the initial condition content vectors for "mao", "group", "Sendero Luminoso" and "bomb." The window is moved one stem at a time

P - 6 1 6 8 F

- 23 -

and this process is continued for each stem in the preprocessed text 702. At each update position, a correction vector is calculated and summed into an accumulated correction vector located in the data items 404 of the stem hash table 400 (shown in Figure 4).

Figure 8 is a diagram illustrating a manipulation process for calculating a stem vector by the vectors of its neighboring stems. Each stem in the stem hash table 400 is modified by the vector of its neighboring stems. For each convolutional window, a vector 800 associated with the center stem is adapted by a vector 802, 804, and 806 which are associated with the stems of the center stem's closest neighbors.

The following equations 1-4 provide the method for calculating the correction for each word in the convolutional window shown in Figure 7.

$$E_{ij} = N_{ij} - T_j \quad (1)$$

The value E_{ij} represents the error or correction between the current target (center) T_j and neighbor N_{ij} . A correction for T_j taking into account all current neighbors in window position is shown by equations 2 and 3.

$$C_j^{\text{CURRENT}} = \sum_i^{\text{WS}} (\|E_{ij}\| - \alpha_{ij}) \hat{E}_{ij} \quad (2)$$

$$C_j^{\text{CURRENT}} = C_j^{\text{ACCUMULATED}} + C_j^{\text{CURRENT}} \quad (3)$$

The variables shown in equations 2 and 3 are defined as follows:

- (i) i = summation dummy variable;
- (ii) C_j^{CURRENT} = correction vector for stem j ;
- (iii) $C_j^{\text{ACCUMULATED}}$ = running summed correction for stem j stored in stem hash table data elements;
- (iv) WS = window size in stems; and
- (v) α_{ij} = a proximity constraint that limits the correcting effect according to the statistics of stem i and j where $\alpha_{ij} = 1 - \Theta_i \Theta_j$.

$$\Theta_i = B + (1 - B) \left(1 - \frac{\log(\frac{1}{ND_i})}{\log(\frac{1}{TND})} \right) \quad (4)$$

Equation 4 is used to calculate the variable α_{ij} for equation 3. The variables shown in equation 4 are defined as follows:

- (i) Θ_i = statistical importance of stem i in the training corpus;
- (ii) B = bounds control parameter determined empirically;
- (iii) ND_i = number of documents that contain stem i; and
- (iv) TND = total number of documents in the training corpus.

Equations 5-9 included below provide a final update to the vectors. It should be noted that the algorithm described in the Caid patent utilizes a batched update process. As such, the algorithm accumulates corrections to the target (center) stem content vector. These corrections are made only at the end of a pass through the training corpus 202. Typically, only two passes are required to achieve excellent learning results. The equations 1-4 are applied for each window position and the results are accumulated. At the end of each pass through all the stems in the intermediate stem file 304, the equations 5-9 are applied. The algorithm disclosed in the equations 5-9 computes the mean stem content vector, subtracts this from each normalized stem content vector and then re-normalizes the entire set of stem content vectors. This computation prevents the stem content vectors from being localized to only one region of the hypersphere and promotes better utilization of the vector space.

$$T_j^{NEW} = T_j^{OLD} + \frac{\gamma}{F_j} C_i^{ACCUMULATED} \quad (5)$$

$$T_j^{NEW} = \frac{T_j^{NEW}}{\|T_j^{NEW}\|} \quad (6)$$

$$M = \frac{1}{NSTEMS} \sum_i^{NSTEMS} T_i^{NEW} \quad (7)$$

$$T_j^{NEW} = T_j^{NEW} - M \quad (8)$$

$$T_j^{NEW} = \frac{T_j^{NEW}}{\|T_j^{NEW}\|} \quad (9)$$

P - 6 1 6 8 F

- 25 -

The variables used in the equations 5 through 9 are defined as follows:

- (i) T_j^{OLD} = old target content vector (initial condition or previous pass);
- (ii) T_j^{NEW} = updated target content vector for current pass;
- (iii) γ = learning rate;
- (iv) F_j = total number of instances of stem j in the training corpus;
- (v) $C_i^{ACCUMULATED}$ = accumulated correction vector for stem i for this pass;
- (vi) M = mean stem vector; and
- (vii) NSTEMS = total number of unique stems in the training corpus.

3. Writes output file.

Once the training is complete, the resulting stem content vectors are written to the word vector database 208.

Session Compiler Module

Figure 9 is a block diagram illustrating the inputs and outputs of the entity vector creation process of the session compiler module 210. In the present invention, the session compiler module 210 performs three functions: text to entity vector transformations, file format conversions, and session control file generation. Each of these functions is described below.

1. Text to entity vector transformations.

To enable the selection of data or information, a text to entity vector representation transformation is required to generate the entity vectors. Figure 9 illustrates the processing performed by the session compiler module 210 when it converts textual information into an entity vector. A preprocessing module 1100 reads the entity descriptions from the entity description database 216. The preprocessing module 1100 is a sub-component of the session compiler module 210 shown in Figure 2. The preprocessing module 1100 module is similar to the preprocessing performed as part of the preprocessing module 300 shown in Figure 3. The preprocessing module 1100 determines if the current word read from the entity descriptions database 216 is on the stop list file 204 or on the phrase list file 206. The valid stems 302 are then used by

P - 6 1 6 8 F

26

an entity vector formation module 1102 to generate static entity vectors which are then stored in the static content vector database 234.

Equation 10 is an example of the process performed by the session compiler module 210 shown in Figure 2 to form entity vectors. The content vector for an entity is the normalized weighted sum of the symbols that comprise that entity. For those entities that are textual descriptions, the description vector is the normalized weighted sum of the stem content vectors of the stems in the description.

$$ECV_i = \frac{\sum_j^{NSTEMS_i} \log(\frac{TND}{ND_j})SCV_j}{\left\| \sum_j^{NSTEMS_i} \log(\frac{TND}{ND_j})SCV_j \right\|} \quad (10)$$

The variables in equation 10 are defined:

- (i) ECV_i = the entity content vector for entity i;
- (ii) SCV_j - the stem content vector for stem j;
- (iii) TND = the total number of documents in the training corpus;
- (iv) ND_j = the number of documents in the training corpus that contain stem j; and
- (v) $NSTEMS_i$ = the number of stems in the description of entity i.

2. File format conversions.

Figure 10a is a block diagram illustrating the process for creating the high speed files 232 shown in Figure 2. The session compiler module 210 performs the file format conversions to facilitate the entity selection process, since the data structure formats that are computationally efficient during word content vector learning are inappropriate for use by the RTAS module 214 during entity selection. The operational flexibility and complexity required for the word content vector learning process is not required by the RTAS module 214, so substantial simplifications are made. As part of this transformation, all of the stem content vectors are pre-scaled by their frequency of

P - 6 1 6 8 F

27

occurrence in the training corpus 202. The stem content vectors are scaled by the same factor shown in equation 10 without the final normalization step. Additionally, the low frequency stems (those that occur less than four times in the training corpus) are removed from the file. The remaining stems represent the statistically most significant portion of the overall stem population. The session compiler module 210 then writes the resulting truncated and pre-scaled stem content vectors to the high speed files 232.

3. Session control file generation.

Figure 10b is a block diagram illustrating the process for generating the session control file 215. The primary function of the session compiler module 210 is the generation of a session control file 215 from operator created session descriptions contained in the session description file 212. The session control file 215 contains the instructions in machine readable format to guide the RTAS module 214 during the ad selection process. The session control file 215 contains the information specified in the session description file 212 and a set of initial entity vectors. The session control file 215 has the ability to identify a group of ads eligible for selection and to provide a set of backup entity vectors in event the EVUS module 228 is not operational. The session information in the session description file 212 describes a set of entities, such as advertisements, that are eligible for selection to become active. The session control file 215 contains a control mechanism for the following attributes for each ad:

- Static or dynamic attribute control. An ad is either statically set to a fixed location in the context vector space or allowed to dynamically adapt based upon user responses.
- Reset. If reset functionality is specified, the entity vector for an existing ad is reset to its initial vector.
- Targeted impression count. The targeted impression count determines the number of impressions that should be delivered for the ad.
- Start date. The start date determines the date and time that the ad is to become eligible for selection.
- End date. The end date determines the date and time that the ad is to become

ineligible for selection.

- Initial conditions. The advertiser can describe the initial entity vector selection for the ad. Options for the initial vector selection include:
 - (i) a vector based upon the ad and the aforementioned text to vector conversion process;
 - (ii) a vector based upon the entity vector of a specified ad; and
 - (iii) a vector based upon an entity vector contained in an external file.

The session compiler module 210 also establishes a connection with the EVUS module 228 (shown in Figure 2) which is the repository for all active entity information. The initial entity vectors are calculated for the ads and the ad identifiers, parameters, and vectors are transmitted to the EVUS module 225 for storage in the entity vector database 230.

Real Time Ad Server Module

The RTAS module 214 provides an Application Programming for Interface (API) local or remote software applications call to perform various ad management routines. The RTAS module 214 selects according to guidelines set in the session control file 215 an advertisement to be displayed to a user.

During the ad selection processing, the RTAS module 214 performs the processing steps listed below.

1. Initialization.

When the RTAS module 214 receives an initialization command, the RTAS module 214 reads the session control file 215 and initializes internal control tables stored in memory. The RTAS module 214 then requests the appropriate entity content vectors as specified by the contents of the session control file 215. The RTAS module 214 requests from the entity vector database 230 via the EVUS module 228 the entity vectors for the active entities and stores them in a memory resident data structure within the RTAS module 214. Once initialization is complete, the RTAS module 214 is ready for real-time operation.

2. Determination of user behavior vector.

When a user action message enters the system via the client API, the RTAS module 214 establishes the behavior vector for that user. If the user is already known to the system, the user id, provided by the user's web browser, is used as the key for a database retrieval. The RTAS module 214 passes the user ID to the PVUS module 224. The PVUS module 224 then returns the profile vector associated with that user from the profile vector database 226. If the user is unknown to the system, a new initialized vector is returned. In either case, the RTAS module 214 has a valid user behavior vector for subsequent operations. This vector is stored within internal memory tables within the RTAS module 214.

3. Observation of user behavior.

The RTAS module 214 also incorporates user behavior into the user's profile vector. These behaviors include a web page request 218 or a search query 217 as shown in Figure 2. The client computer 219 passes a behavior packet including the observed behavior in the client API. The RTAS module 214 uses the information contained in the user behavior packet to generate a behavior vector for the user action.

Figure 11 is a block diagram illustrating the inputs and outputs of the process for transforming a page request into an content vector which is performed by the RTAS module 214 shown in Figure 2. If the observed user behavior includes a page, navigation or hierarchy request 218, a page content vector lookup module 1402 which is part of the RTAS module 214 retrieves the entity vector associated with the ID of the requested page from the static content vector database 234. The RTAS module 214 augments the entity vector module of the user requested page with the user ID to provide a behavior vector.

Figure 12 is a block diagram illustrating the inputs and outputs of the process for transforming a query into a content vector which is performed by the real-time ad server module shown in Figure 2. If a client computer 219 observes user behavior such as a user query 217, the RTAS module 214 transforms the query 217 into a content vector by using the memory image of the high speed file 232 to provide an entity vector based upon the query terms. A preprocessing module 1502 which is part of the RTAS module

214, removes common words that are identified in the stop list file 204, combines the words that are part of a phrase list file 206 and performs the stemming process for all of the terms in the user query 217. The RTAS module 214 then creates a user behavior vector by augmenting the result of the aforementioned query 217 with the user ID by a form entity content vector module 1504.

4. Selection of best entity to present.

Figure 13 is a block diagram illustrating the functions of the ad selection process performed by the RTAS module 214 shown in Figure 2. If the client computer 219 (shown in Figure 2) does not observe user behavior, the RTAS module 214 uses historical user profile vector from the profile vector database 226 as the basis for selection of the best entity to present. If the client computer 219 observes interactive behavior, the RTAS module 214 uses the observed user behavior vector as the basis for selecting the best entity to present.

The RTAS module 214 uses the historical user profile vector or behavior vector as the basis of a vector comparison against all active ads as specified by the session control file 215. The contents of the session control files 215, which are loaded into memory by the RTAS module 214 during initialization, are available for high speed searches. By placing the session control files in memory of the RTAS module 214, a performance gain is realized.

Referring again to Figure 13, the RTAS module 214 performs the optimum entity selection by performing dot products of entity vectors with behavior and user vectors (state 1600). The RTAS module 214 performs a search process by computing a vector closeness metric between the vector to be compared and all active ad vectors. Although the RTAS module 214 may use any vector proximity metric, the preferred embodiment of the invention uses a vector inner product.

At a state 1602, the RTAS module 214 selects the closest ad subject to inventory management constraints. Specifically, if the closest ad has already been presented sufficient times to the user population, the RTAS module 214 presents the second closest matching ad is selected. The inventory management scheme can be arbitrarily

P - 6 1 6 8 F

- 31 -

complex and the RTAS module 214 has provisions to interface to an external inventory management module, that may be used to adaptively balance ad presentations subject to a series of external constraints. An inventory weighting module which is a sub-component of the RTAS module 214 performs all of the inventory control operations by adjusting the vector closeness weights. The RTAS module 214 then measures and ranks each ad ID by score (state 1604).

Once the closest entity or ad has been selected, subject to the constraints described above, the RTAS module 214 passes the ID of the selected ad via the API to the client computer 219. The ID is used by the client computer 219 to retrieve and display the actual ad image to the user via the web.

To improve performance, the RTAS module 214 caches the set of entity vectors and parameters described by the active session control file 215. The RTAS module 214 obtains periodic updates from the EVUS module 228 for changed entities.

5. Observe user response to entity presentation.

Once the best entity or ad is determined and the host system generates a web page that incorporates the selected ad, the user response to the ad is monitored. If the user requests more information on the product, as indicated by clicking on the entity or ad, this event is sent via the API to the RTAS module 214. However, all ad presentations need not result in a message to the RTAS 214 module.

6. Update of user behavior vector.

The RTAS module 214 adapts the user behavior profile vector to reflect observed user actions. The observed user behavior vector is then used to update the profile vector for that user. The RTAS module 214 sends the user ID and observed user behavior vector to the PVUS module 224 for modification of the historical user profile vector. The operation of the PVUS module 224 is described below in greater detail.

7. Update of ad or entity vector.

If the user responds to the entity (ad, coupon or solicitation), the entity vector and user ID are sent to the EVUS module 228 for update (shown in Figure 17). The EVUS module 228 then adapts the entity vector toward the observed behavior vector of

P - 6 1 6 8 F

- 32 -

the user. The operation of the EVUS module 228 is described below in greater detail.

Profile Vector Update Server Module

The PVUS module 224 is responsible for the storage, retrieval and updating of the user profile vectors that are contained in the profile vector database 224.

These user profile vectors are the basis for a wide range of processing capabilities including:

- (i) dissemination of targeted advertising, coupons, offers and information to the user;
- (ii) analysis of user behavior;
- (iii) correlation of user behavior with advertising, product and service descriptions;
- (iv) inference of user characteristics based on cross-correlation of user behavior profiles with demographic purchasing databases;
- (v) tracking user behavior over time;
- (vi) predictive modeling of likely user behavior and optimization of information presentation based on these predictive models; and
- (vii) accumulation of effectiveness and click-through statistics to optimize predictive models, provide management reports and lay groundwork for increased ad revenues.

In the preferred embodiment of the invention, the PVUS module 224 also incorporates external information, such as credit card purchasing histories into the user profiles. This information is easily incorporated, since the content vector representation accommodates the aggregation of heterogeneous symbolic information, such as web behavior and credit card purchases, into a unified high dimensional space.

The PVUS module 224 is implemented as one or more executable computer programs that receive requests from clients over a network protocol such as TCP/IP and generates responses to such requests. The PVUS module 224 has one or more physical databases which are used to store the user profile content vectors and parameters which

P - 6 1 6 8 F

- 33

are indexed by unique user IDs. Multiple PVUS modules may be in simultaneous operation to handle large numbers of users. When multiple servers are used, they are addressed by the clients as a single entity. The PVUS modules may be scaled across multiple hardware platforms to support a large user population. This distribution is accomplished by multiplexing the user ID to select one of N physical servers. Each server may further multiplex the request to one of M physical databases. Multiplexing is achieved by generating a hash code from the user ID which results in essentially random bit distributions in the hash code. There are many multiplexing hashing algorithms, that may be used. However, the preferred embodiment uses the MD5 algorithm. The multiplexor then uses a modulo N on the hash code to determine which of N children should handle the request.

The PVUS module 224 performs the processing actions listed below.

1. Creation of new user profile vectors.

Each user in the system is assigned a unique user ID. In the web environment this could be the user's Internet IP address or a host selected ID placed in the browser "cookie" file. If a request is made for a user ID that is not contained in the profile vector database 226, the PVUS module 224 creates an entry for the new user and initializes the new user profile vector to a pseudorandom content vector which will be later adjusted by the adaptation algorithm (shown in Figures 19a and 19b).

2. Retrieval of user profile vectors and parameters.

The PVUS module 224 obtains a user profile vector by using the unique user ID to determine a hash code. The entry in the hash table at that location contains a pointer to the information in the profile vector database 226. The PVUS module 224 uses the database pointer to perform the retrieval of the user profile vector. If the request for the profile vector is from the RTAS module 214, the PVUS module 226 passes the profile vector to the RTAS module 214 for subsequent use.

3. Adaptation of user profile vectors.

The PVUS module 224 performs the adaptation of the user profile vectors based upon some observation about a user's behavior. As the user traverses a web site

utilizing the present invention, behaviors are observed including user requested queries, web pages and ads. Each observed behavior is converted into a content vector and is also used to adapt a user's profile vector which is stored in the profile vector database 226.

Figure 14 is a block diagram illustrating the profile vector adaptation process of the PVUS module 224 shown in Figure 2. The PVUS module 224 contains several sub-components including a transform query to content vector (TQCV) module 1700, a transform action to content vector (TACV) module 1704, a profile vector lookup (PVL) module 1706, and a profile vector adaptation (PVA) module 1702. If the user enters a query 217 to the system, the query string and the user ID are passed to a TQCV module 1700. The TQCV module 1700 transforms the query string into a query content vector in the same manner as the process shown in Figure 10b. The resulting query content vector is passed to the PVA module 1702 for adaptation of the user profile vector. The PVA module 1702 utilizes the adaptation algorithm described below and shown in Figures 19a and 19b. A profile vector database update module 1704 writes the updated profile vector to the profile vector database 226 for later use.

If the user clicks on an ad or purchases a product 222, the TACV module 1704 uses the ad or product description to adapt the user profile vector. This module 1704 transforms the entity ID into a content vector via a table lookup in memory within the PVUS module 224. In parallel, the user ID is used to lookup the user profile vector from the profile vector database 226. The PVA module 1702 inputs these two vectors into the adaptation algorithm described in Figures 19a and 19b. The profile vector database update module 1704 then writes updated profile vector to the profile vector database 226 for later use.

4. Demographic predictions and tagging.

For many advertising and sales applications, it is advantageous to have an estimate of the demographic parameters for individuals as well as the entire user population. In the present system, although no user registration is performed, advertisers can analyze the profiles vectors to retrieve user demographics. Advertisers

can use these profile vectors to estimate the age, gender, household income (HHI), and other demographic parameters of a user. The present system has provisions to incorporate demographic models into its operation. When using these demographic models, the demographic parameters of the entire population of users are estimated and used to augment the information in the profile vector database 226. This estimate process can be performed by both conventional statistical methods or via neural network techniques.

Figure 15 is a block diagram illustrating the inputs and the outputs of the model building process performed by the relationship, discovery and analysis module shown in Figure 2. Predictive models for the estimation of age, gender and HHI are constructed using a combination of previous observed behavior contained in a historical user actions database 226, ground truth information from user registration on a different service or via a ground truth user demographics database 1904, and profile vectors from the profile vector database 226. Using this information, a modeling module 1900 constructs a set of model parameters. These model parameters are stored in a model parameter database 1906. Any statistically valid process can be used in the modeling module 1900, including neural network techniques, to derive the predictive model parameters. The typical outputs of the predictive model are age, gender and HHI.

Figure 16 is a block diagram illustrating the process by which the PVUS module 224 updates the profile vector database 226 with demographic information. For each user in the profile vector database, the PVL module 1706 retrieves the user's associated profile vector from the profile vector database 226. The PVL module 1706 passes this information, with the user ID and historical actions to a predictive model processing module 2000. The predictive model processing module 2000 uses the model parameters from a model parameter database 1906 in conjunction with the current user profile vector and previous observed behavior to form an estimate of a set of demographic parameters for the current user. This estimate is added to the information contained in the profile vector database 226 by a profile vector update module 1704. The augmented contents are used as input to the RTAS module 214 during ad selection and as part of

the relationship discovery and analysis process.

Entity Vector Update Server Module

The EVUS module 228 is responsible for the storage, retrieval and updating of the entity profile vectors from the entity vector database 230. The EVUS module 228 is the main processing module in the system to read and to write the entity vector database 230.

The EVUS module 228 consists of one or more executable computer programs that receive requests from clients over a network protocol (TCP/IP) and generates responses to such requests. The EVUS module 228 may have one or more physical databases which are used to efficiently store and index the user profile content vectors and parameters by unique entity IDs.

During normal operation, the EVUS module 228 performs the following processing.

1. Entity insertion and deletion.

The EVUS module 228 inserts new entities or ads and removes old entities or ads from the entity vector database 230 in response to commands from the session compiler module 210. When a new session is compiled, the session compiler module 210 sends update commands, new entity vectors and parameters to the EVUS module 228. Additionally, the session compiler module 210 sends commands to the EVUS module 228 to delete inactive entities from the entity vector database 230.

2. Store and retrieve entity vectors and parameters.

The EVUS module 228 stores and retrieves the current parameters and vectors for all the active entities in the system and retrieves the entity vector and associated parameters for a given entity ID. During the initialization of the RTAS module 214, the RTAS module 214 sends a command to the EVUS module 228 requesting a copy of the most recent entity vectors and parameters to hold in the memory of the RTAS module 214.

3. Entity profile vector adaptation.

P - 6 1 6 8 F

- 37 -

Figure 17 is a block diagram illustrating the entity vector adaptation process of the EVUS module 228 shown in Figure 2. The EVUS module 228 performs the entity vector adaptation in response to a message indicating that a user has responded to (clicked on) an entity. The EVUS module 228 includes various sub-components including an entity vector adaptation module 2100, a transform query to content vector (TQCV) module 2102, an entity vector lookup module 2104, a profile vector lookup module 2106 and an entity vector database update module 2108.

An adaptation event occurs when a user enters a query 217 or clicks on an ad or performs an on-line purchase of a product 222. When the entity adaptation event occurs, the user action that resulted in the event, such as a user query 217, the EVUS module 228 obtains from the RTAS module 214 the entity ID for the selected ad and the user ID.

If the user requests a query 217, the TQCV module 2102 transforms the actions into a query content vector, and passes the result to the entity vector adaptation module 2100. If the user event was a clickthrough or purchase 222, the entity vector lookup module 2104 retrieves the entity vector for the selected entity from the entity vector database 230. The entity vector lookup module 2104 passes the vector for the selected entity to the entity vector adaptation module 2100. If the user ID was passed into the EVUS module 228, the profile vector lookup module 2106 retrieves the profile vector for the user from the profile vector database 226. The resulting content vector is then passed to the entity vector adaptation module 2100.

After receiving the inputs from the TQCV module 2102, the entity vector lookup module 2104, and the profile vector lookup module 2106, the entity vector adaptation module 2100 performs the equations shown in Figures 19A and 19B. The entity vector adaptation module 2100 then passes the updated entity vector along with the entity ID and time stamp to the entity vector database update module 2108 where the information is stored in the entity vector database 230.

4. Time updates.

The EVUS module 228 generates a list of entity IDs whose vectors or

P - 6 1 6 8 F

38

parameters that have changed since a given moment in time. This information is used to refresh the memory image of the EVUS module 228 that is maintained by the RTAS module 214. When the refresh command containing a time stamp of the last update is sent from RTAS module 214, the EVUS module 228 sends to the RTAS module 214 the entity IDs, vectors parameters for those items that have changed and a new time stamp. This information is then saved in the RTAS internal memory for use in the entity selection process.

5. Inventory management.

The EVUS module 228 performs inventory management weighting across all the ads in the system. The EVUS module 228 uses a weight parameter to adjust each of the ad selection scores. The weight parameter is determined by the ads targeted number of impressions and the time remaining for each ad.

Profile and Entity Adaptation Algorithms

Figure 18 is a flowchart illustrating the vector adaptation process of the profile vector update server and the entity vector update server upon the receipt of a user query 217 (shown in Figure 2). Both the PVUS module 224 and the EVUS module 228 apply a set of adaptation algorithms to allow the evolution of profile vectors and entity vectors as a result of user interaction with the system.

Starting at a state 2250, the PVUS module 224 updates the profile vector based upon the current user query 217. Moving to a decision state 2252, the profile vector of the user and the behavior vector for the search query is compared to the entity vectors and ranked according to this comparison to establish a score. The profile score of the user is compared against the behavior score for the search query to determine which score is larger. If the query score is larger, the PVUS module 224 proceeds to update the bias vectors (state 2254). The bias vectors are used to make sure that during the continual mutual adaptation process that the vectors do not all end up with the same entity vector. Next, at a state 2258 an advertisement is selected based upon the query score.

P - 6 1 6 8 F

39

Proceeding to a decision state 2260, the system checks whether a user clicked on the provided advertisement. If the user did not select the advertisement the system returns to the state 2250 to repeat the adaptation process upon future user queries. If the user selected an advertisement, the EVUS module 228 updates the selected ad vector with the behavior vector for the user query 217 (state 2262). Next, at a state 2264, the PVUS module 224 updates the profile vector with the entity vector for the advertisement and moves to a state 2274.

Referring again to the decision state 2252, if the profile score is greater than the query score, the bias vectors are updated in state 2265. Next, at a state 2266, the ad selected by the profile scoring is sent to the client 219. Proceeding to a decision state 2268, the system checks whether a user clicked on the provided advertisement. If the user did not select the advertisement, the system returns to the state 2250 to repeat the adaptation process upon the receipt of future user queries 217. If the user selected an advertisement, the PVUS module 224 updates the profile vector with the entity vector for the advertisement (state 2270). The EVUS module 228 then updates the selected ad vector with the profile vector for the user query 217.

Moving from a state 2264 or a state 2272, at a state 2274 the cost model associated with the selected advertisement is updated. The system then returns to the state 2250 to repeat the adaptation process upon the receipt of future user queries 217.

The profiles of individual users and the corresponding current query are represented by the following two sets shown in equations 11 and 12.

$$P_{uid} = \{p_1^{(uid)}, \dots, p_d^{(uid)}\} \quad (11)$$

$$Q_{uid} = \{q_1^{(uid)}, \dots, q_e^{(uid)}\} \quad (12)$$

User queries includes topical information which may be represented by more than one content vector. Equation 13 shows that each ad is represented in a similar way to the profile vectors.

$$A_{adid} = \{a_1^{(adid)}, \dots, a_f^{(adid)}\} \quad (13)$$

Equations 14 and 15 are applied during the decision state 2252 shown in Figure 15 when computing the scores.

P - 6 1 6 8 F

- 40 -

$$p_score(P_{uid}, A_1, \dots, A_a) = \max_{i,j, adid} [p_i \cdot a_j^{(adid)} Value_{adid}] \quad (14)$$

$$q_score(Q_{uid}, A_1, \dots, A_a) = \max_{i,j, adid} [q_i \cdot a_j^{(adid)} Value_{adid} \times I(q_i \cdot a_j^{(adid)} > threshold_{query})] \quad (15)$$

The *Value* parameter is computed by traditional cost model algorithms (not shown).

Figures 19a and 19b are the equation flow diagrams for the fixed rate learning algorithm performed by the profile vectors update module and the entity vector update module shown in Figure 2. Unless otherwise noted, the adaptation equations shown in Figures 19a and 19b are common to both profile and entity content vector updates. The parameters used to control the adaptation process are shown in Figure 20.

The following is a description of when the equations illustrated in Figures 19a and 19b are performed in relation to the states illustrated in Figure 18. The equations in a block 2276, 2284 (shown in Figure 19a) are performed during the state 2250 (shown in Figure 18) during the process of updating the profile vector with a query. The equations in a block 2278 are performed at the state 2262 during the process of updating the ad vector with a user query after the user selected an advertisement. Similarly, the equations in block 2280 are performed at the state 2264 during the update of the profile vector with the selected advertisement.

The equations in block 2282 (shown in Figure 19a) are performed at the state 2254 (shown in Figure 18) to update the bias vectors to insure that all of the vectors do not end up with the same multidimensional vector. The equations in a block 2286 are performed at the state 2272 during the update an ad vector with the profile vector. These equations are performed when the profile score of the user is greater than the query score. Similarly, the equations in a block 2288 (shown in Figure 19b) are performed in the state 2270 (shown in Figure 18) during the update of the profile vector with the ad vector.

The equations in a block 2290 (shown in Figure 19b) are performed during the update of the bias vectors after the profile score of the user is greater than the score for the users query at the state 2265 (shown in Figure 18).

P - 6 1 6 8 F

- 41 -

Calculation of Entity Inventory Weighting (Cost Model)

The RTAS module 214 keeps track of the number of impressions generated for each entity or ad and periodically sends a message to the EVUS module 228 containing each ad ID and the number of impressions generated for each ad. The RTAS module 214 uses the ad parameters specifying the targeted number of impressions and the time each ad will remain eligible for serving to calculate inventory management weights. The weights are applied by the RTAS module 214 when scoring the active ads. The RTAS module 214 causes under-impressed ads to receive more impressions while suppressing impressions for over-impressed ads.

Relationship Discovery and Analysis Processing Module

The present system has the ability to perform relationship discovery and analysis on the contents of the profile vector database 226 and the entity vector database 230 (shown in Figure 2). The results of the analysis process are listed in a series of reports that are presented to an analyst 240. These reports provide high value actionable information about the behavior pattern of the system users.

The RDA module 236 takes the contents of the profile vector database 226 and the entity vector database 230 to produce the analysis reports 238 (shown in Figure 2). A variety of conventional statistical techniques or neural network approaches are utilized to perform this analysis.

The RDA module 266 performs three analytical functions including unsupervised categorization, supervised categorization and conventional statistics. Each of these functions is described below.

1. Unsupervised categorization.

The RDA module 236 performs unsupervised categorization or clustering of the contents of the profile vector database 225, the entity vector database 230 or both of these modules. The RDA module 236 may use any clustering algorithm as long as the algorithm can process real number vectors. The result of the clustering is a set of cluster centroid vectors that represent regions of high density. These centroids are the same

P - 6 1 6 8 F

- 42 -

type of vectors that are used for all the other processing operations. Thus, the RDA system is able to find a set of word vectors that are closest in the content vector space to the centroid vector. The words associated with these vectors, and their closeness metrics, constitute the key concepts of the centroid. The evolution of cluster centroids over time provides trend information. The identification of word vectors close to these cluster centroids provides a key word summary for each cluster, easing the analysis and interpretation task. Conventional statistical analysis may also be used to investigate cluster contents. The variance of cluster membership is a reliable indication of cluster coherence and "unity of focus" in terms of behavior. The cluster identified in the product vector database 226 provides highly valuable information about the characteristics of the users who purchase the products. This result is a natural consequence of the mutual adaptation between the ad vectors and the user content vectors. The number of clusters found and the explanation of the clusters are listed in the analysis reports 238 for inspection by the analyst 240.

2. Supervised categorization.

The RDA module 236 also performs supervised categorization of the contents of the profile vector database 226, the entity vector database 230, or both of these modules. For supervised categorization, the RDA module 236 may employ any algorithm that can process real number vectors. The result of the categorization is a set of users and/or entities that are members of each category. The count and membership list, by category, are listed in the analysis reports 238 for inspection by the analyst 240.

3. Conventional statistics.

The RDA module 236 applies conventional statistics to the contents of the profile vector database 226, the entity vector database 230, or both of these modules. This analysis produces counts, count distributions, means and variances of the database contents as a whole or any subset which is selected by the analyst 240. The results of the processing are listed in the analysis reports 238 for inspection by the analyst 240.

A summary of the operation of the system for the advertising scenario is as follows.

P - 6 1 6 8 F

43

System Initialization

First, the word content vector learning module 200 creates multiple content vectors from text input. This technique, results in a set of word content vectors, one word content vector for each word found in the training vocabulary and not in the stop list 204. In the advertising scenario, the training texts are usually product descriptions and trade magazines. For instance, if the targeted domain of operation is sporting goods, the training corpus would contain the product descriptions from sporting goods catalogs, the contents of several years of "Field and Stream" magazine and books on outdoor life. The selection of the correct training corpus, oriented toward the domain of operation, such as outdoor sports, gives the correct "spin" on the usage of words. In the aforementioned example, "bow" becomes associated with "arrow" as opposed to "violin" or an action performed on stage at the completion of a performance.

Next, the session compiler module 210 computes an initial set of entity vectors from descriptions of the products for sale. These descriptions might come from the product catalog or be typed in by a human operator. For instance, "Warm Goretex, down filled parka perfect for those cold winter nights and days on the ski slopes" might serve as the text initialization for the entity vector for a coat. The entity vector is a normalized weighted sum of the word vectors of the words in the description. The session compiler module 210 creates an initial entity vector for each ad in the system.

The session description file 212 describes the guidelines for ad placement, such as display times authorized, sites, which ads to show and how many impressions are available, is compiled into a machine usable form. This results in the creation of a session control file 215 with information that controls the operation of the ad serving process. Finally, a plurality of user profile vectors are initialized to NULL.

Regular System Operation

During normal operation, an Internet user visits the web site and the user's identity is established via login, or through the web browser cookie. Each user identity is associated with a profile vector. If the user is new, a null profile vector is created and

P - 6 1 6 8 F

- 44 -

added to the profile vector database. If the user is known to the system, the user's profile vector is retrieved from the profile vector database.

The system then observes the user's behavior. If a user performs a search query 217, the user profile vector is updated using the content vector of the current query. If the user requests a web page to read 218, the user profile vector is modified by the content vector of the requested web page.

The updated profile vector is then used as the basis for selecting relevant ads to display to the user. Relevance is determined by closeness of the user profile vector to entity vectors stored in the ad vector database. The closest entity vector is selected as being most relevant.

If the user is presented an ad, and the user requests more information of the subject of the ad by clicking on the ad, both the current ad entity vector and the profile vector are updated. The user profile vector is moved or adapted a small step in the direction of the entity vector. The entity vector is moved or adapted a small step in the direction of the profile vector of the person who clicked on the ad. Thus, the system allows ads to assume the characteristics of the users that purchase them in a real-time adaptive manner.

Discovery and Analysis Mode

During discovery and analysis mode, at periodic intervals, the system administrator performs relationship discovery and analysis of the contents of both the entity vector database 230 and the user profile vector database 226 to discover useful and exploitable characteristics of user behavior. The RDA module 236 performs vector clustering, reporting and provides summary statistical information of system effectiveness in terms of user clickthrough rate. The behavior clusters discovered during analysis serves as the basis of advertising campaigns and provides valuable insight into user behavioral preferences. The behavior clusters can be utilized as the initial conditions for entity vectors the advertiser wants to target by group, rather than on an individual basis.

The system shown in Figure 2 has two sets of adaptive components. First are the user profile vectors. Second are the content vectors for each of the advertisements. Both sets of vectors influence each other based upon the actions of the users. Ads are updated in such a manner as to adapt toward regions of high user interest, thus improving system effectiveness and automatically discovering group preferences. User profile vectors adapt based on the observed actions of the user, thus providing an accurate and timely representation of user preferences both individually and as a group.

Since the sets of vectors are dependent on a user's actions and evolve with time, they encode a wealth of information about the behavior of the user population. These content vector sets can be "mined" with the analysis tools to provide valuable actionable insight concerning both individual, group and whole population behavior.

While the above detailed description has shown, described, and pointed out the fundamental novel features of the invention as applied to various embodiments, it will be understood that various omissions and substitutions and changes in the form and details of the system illustrated by those skilled in the art, without departing from the intent of the invention.

4 Brief Description of Drawings

Figure 1 is a functional block diagram illustrating a computer environment that may be associated with the present invention.

Figure 2 is a functional block diagram illustrating the processing and data components of the present invention.

Figure 3 is a block diagram illustrating the inputs and outputs of the pre-processing process of the word content vector learning module shown in Figure 2.

Figure 4 is a block diagram illustrating the inputs and outputs of the hash table creation process of the word content vector learning module shown in Figure 2.

Figure 5 is a block diagram illustrating the functions of the stem generating process of the word content vector learning module shown in Figure 2.

Figure 6 is a block diagram illustrating the stem frequency statistics calculation

process of the word content vector learning module shown in Figure 2.

Figure 7 is a diagram illustrating the different learning convolutional windows of the word content vector learning module shown in Figure 2.

Figure 8 is a diagram illustrating the vector manipulation process for adapting a stem vector by the vector of neighboring stems for a convolutional window shown in Figure 7.

Figure 9 is a block diagram illustrating the inputs and outputs of the entity vector creation process of the session compiler shown in Figure 2.

Figure 10a is a block diagram illustrating the inputs and outputs of the process for creating the high speed files shown in Figure 2.

Figure 10b is a block diagram illustrating the inputs and outputs of the process for generating the session control file shown in Figure 2.

Figure 11 is a block diagram illustrating the inputs and outputs of the process for transforming a page request into an content vector which is performed by the real time ad sever module shown in Figure 2.

Figure 12 is a block diagram illustrating the inputs and outputs of the process for transforming a query into a content vector which is performed by the real time ad server module shown in Figure 2.

Figure 13 is a block diagram illustrating the functions of the ad selection process performed by the real time ad server shown in Figure 2.

Figure 14 is a block diagram illustrating the functions of the profile vector adaptation process of the profile vector update server shown in Figure 2.

Figure 15 is a block diagram illustrating the inputs and the outputs of the model building process performed by the relationship, discovery and analysis module shown in Figure 2.

Figure 16 is a block diagram illustrating the functions of the process by which the profile vector update server module shown in Figure 2 updates the profile vector database with demographic information.

Figure 17 is a block diagram illustrating the functions of the process by which

P - 6 1 6 8 F

47

the entity vector update server module shown in Figure 2 adapts entity vectors.

Figure 18 is a flowchart illustrating the vector adaptation process of the profile vector update server and the entity vector update server shown in Figure 2.

Figures 19a and 19b are the equation flow diagrams for the fixed rate learning algorithm performed by the profile vectors update module and the entity vector update module shown in Figure 2.

Figure 20 is a chart containing a summary of sample run time parameters for the fixed rate learning algorithm shown in Figures 19a and 19b.

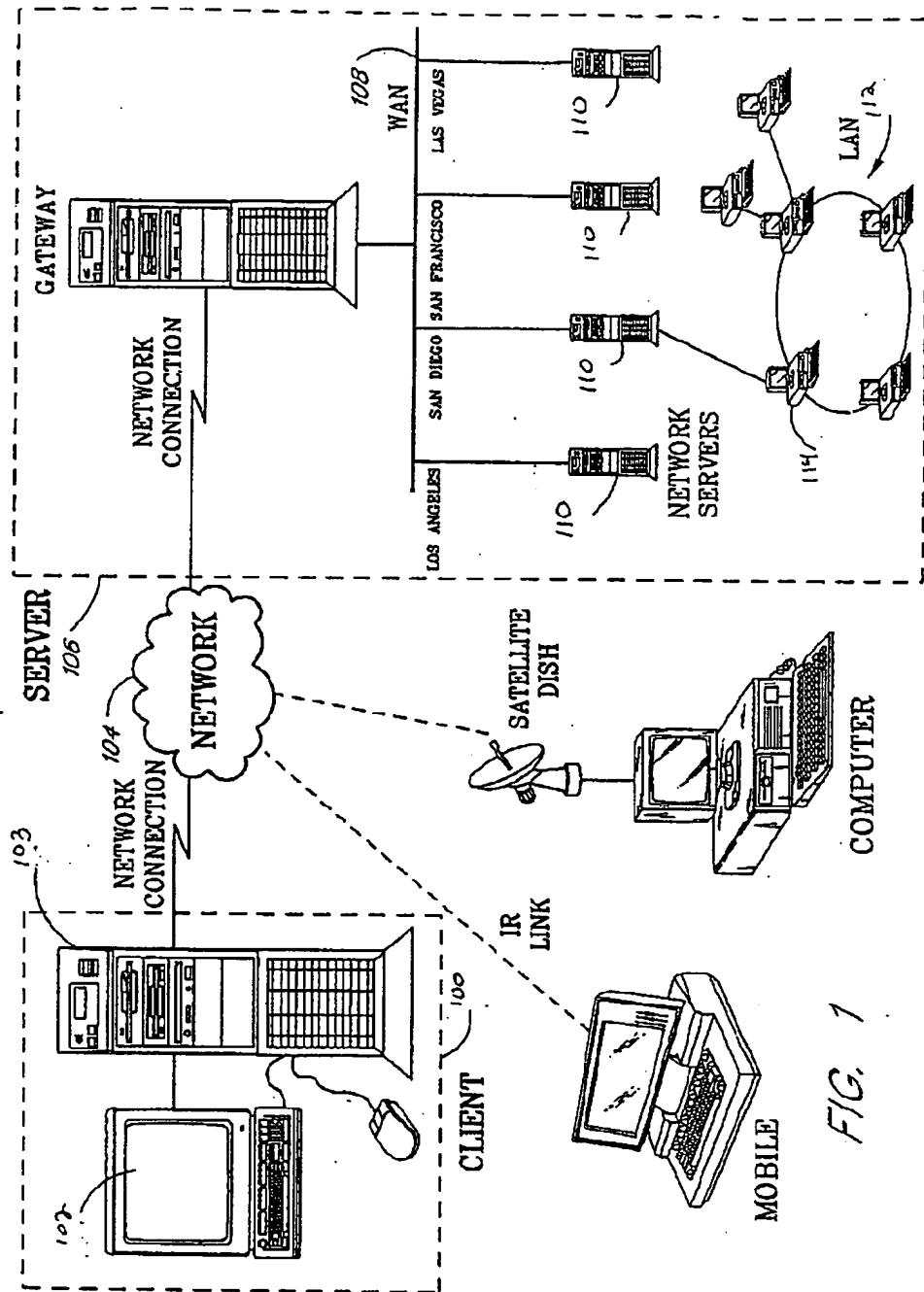


FIG. 1

P - 6 1 6 8 F

- 2 -

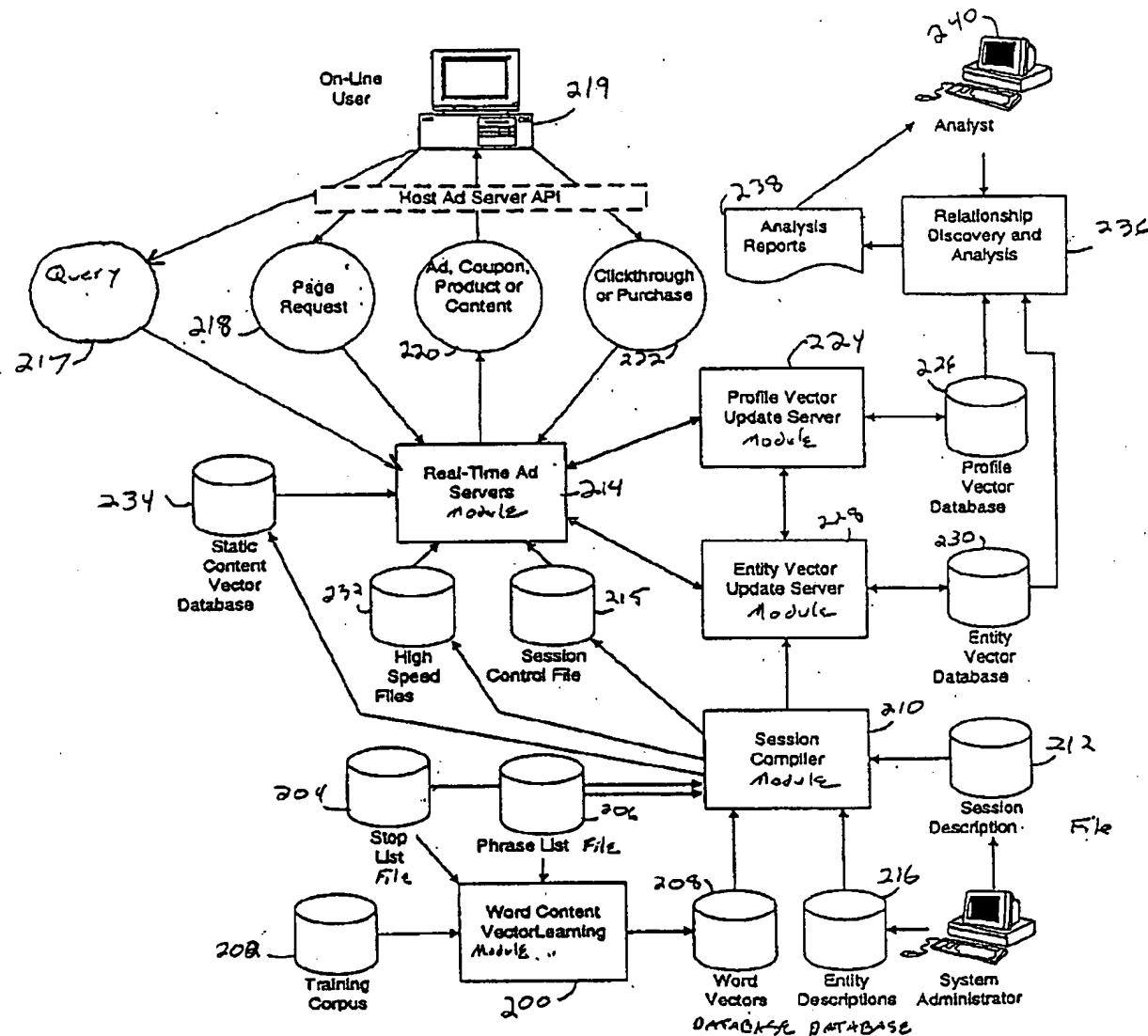


Figure 2

P - 6 1 6 8 F

- 3 -

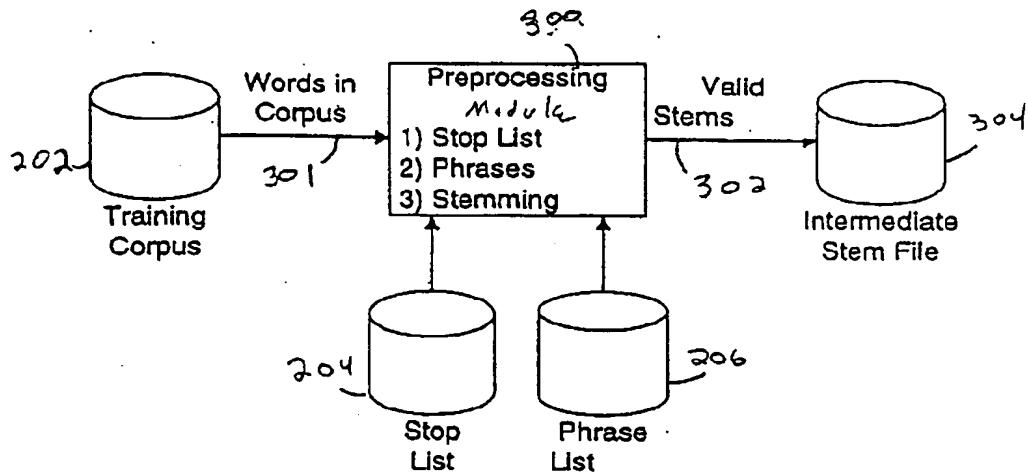


Figure 3

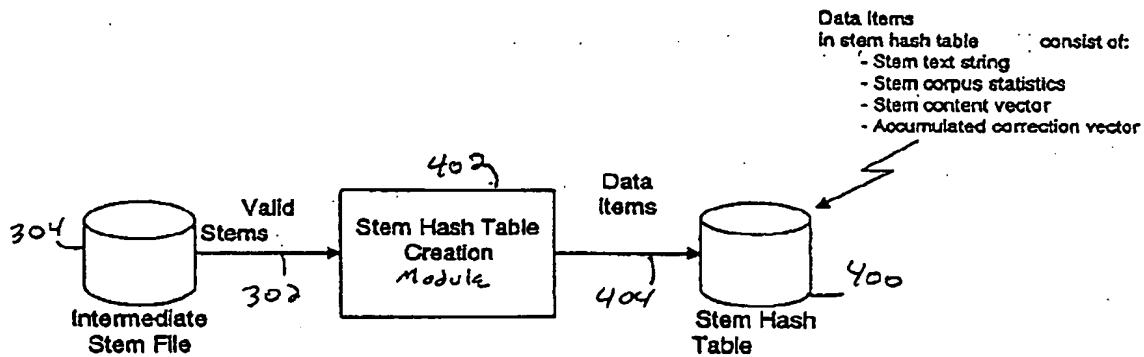


Figure 4

P - 6 1 6 8 F

- 4 -

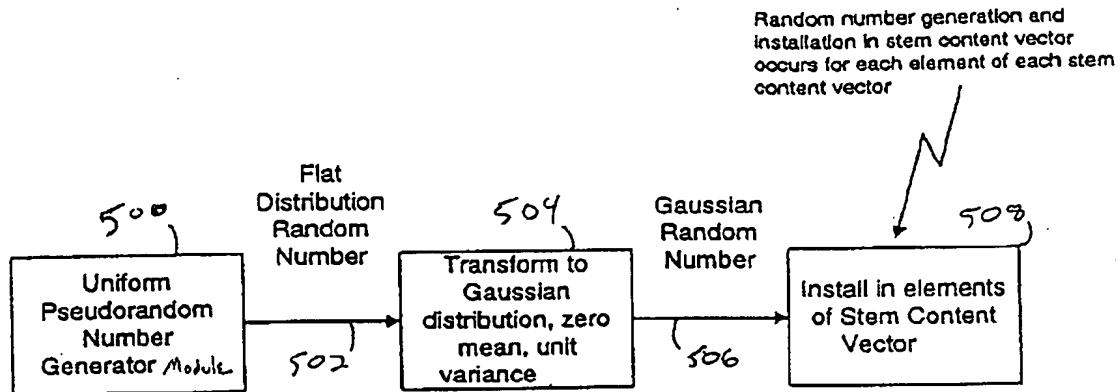


Figure 5

Statistics computed for each valid stem A11 are:
 1) Total occurrences in corpus
 2) Total number of documents that contain stem

Data items in stem hash table () consist of:
 - Stem text string
 - Stem corpus statistics
 - Stem content vector
 - Accumulated correction vector

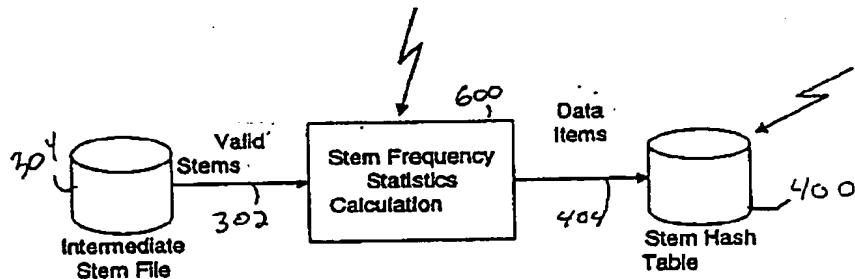


Figure 6

P - 6 1 6 8 F

- 5 -

700 Source Text

The Maoist rebel group Sendero Luminoso bombed the DEA office in Lima, Peru.

702 Preprocessed Text

Mao rebel group Sendero Luminoso bomb DEA offic Lima Peru

704 Convolutional Window # 1

Mao rebel group Sendero Luminoso bomb DEA offic Lima Peru
Center
Neighbor

706 Convolutional Window # 2

Mao rebel group Sendero Luminoso bomb DEA offic Lima Peru
Center
Neighbor

Convolutional Window # 3

Mao rebel group Sendero Luminoso bomb DEA offic Lima Peru
Center
Neighbor

Convolutional Window # 4

Mao rebel group Sendero Luminoso bomb DEA offic Lima Peru
Center
Neighbor

Figure 7

P - 6 1 6 8 F

- 6 -

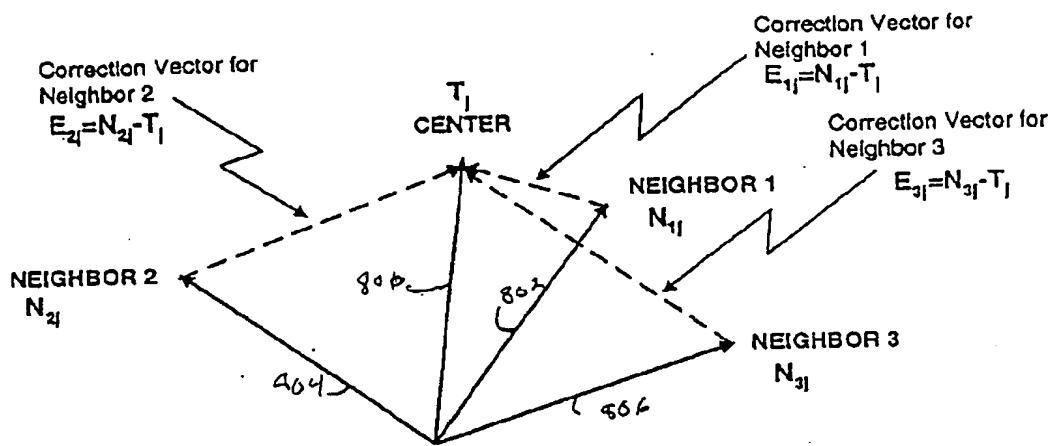


Figure 8

P - 6 1 6 8 F

- 7 -

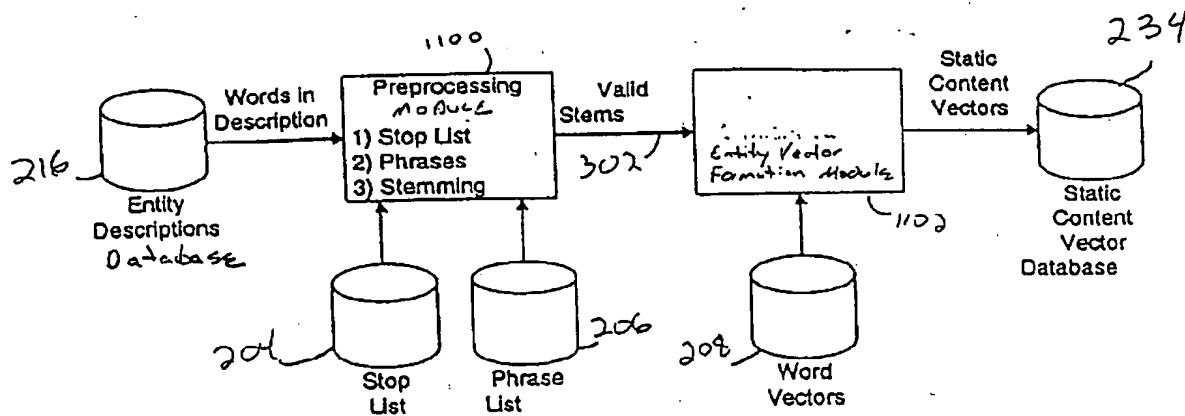


Figure 9

P - 6 1 6 8 F

- 8 -

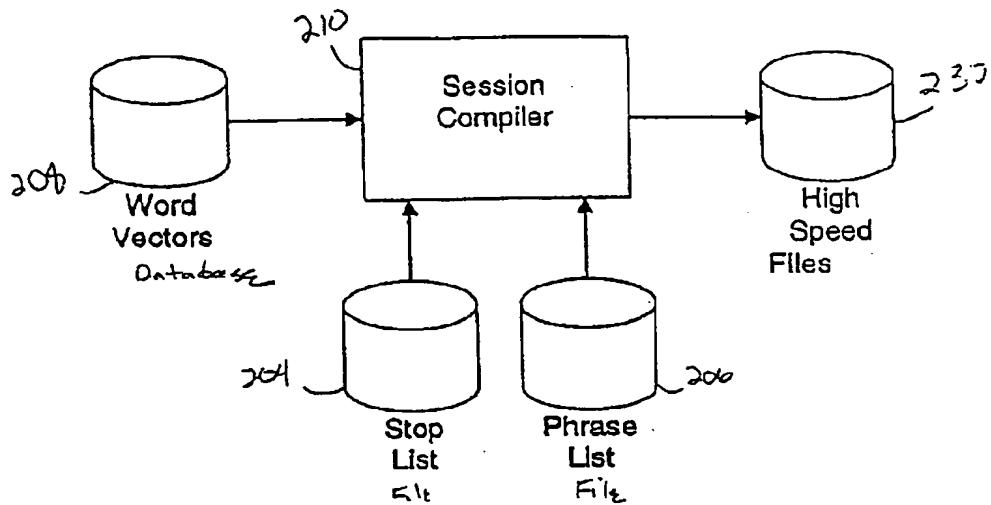


Figure 10a

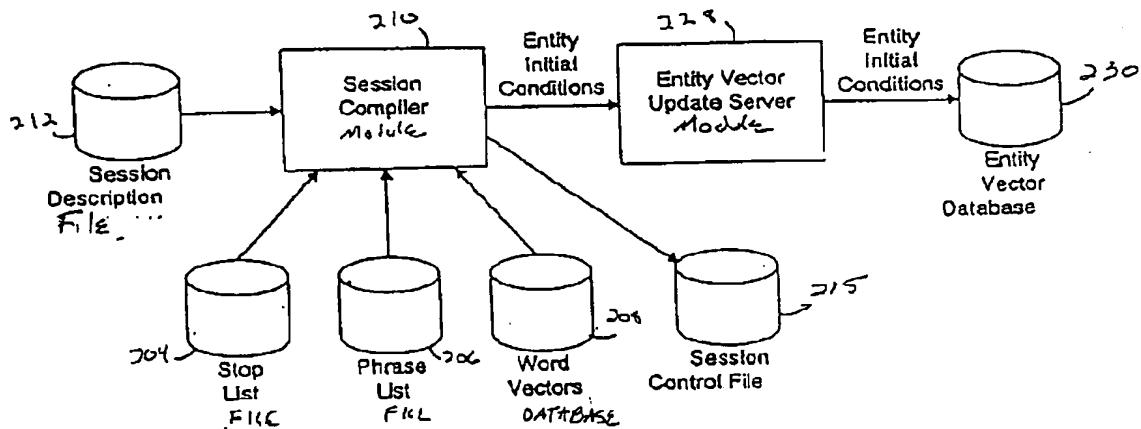


Figure 10b

P - 6 1 6 8 F

- 9 -

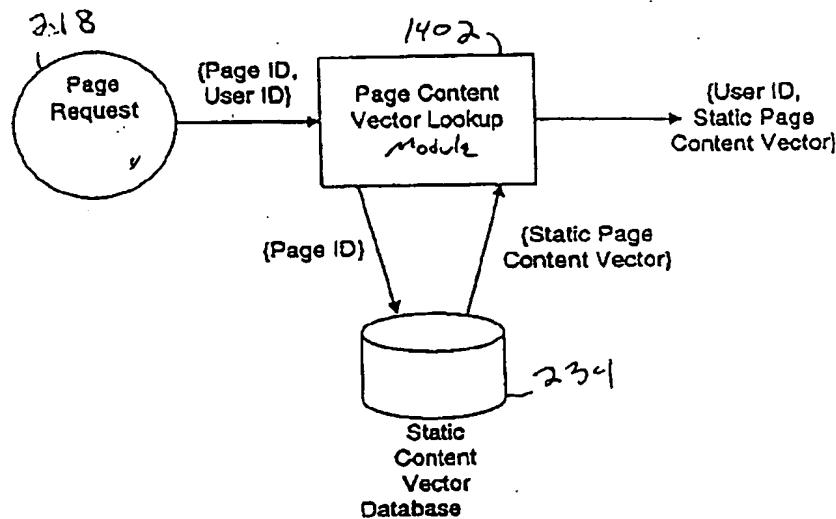


Figure 11

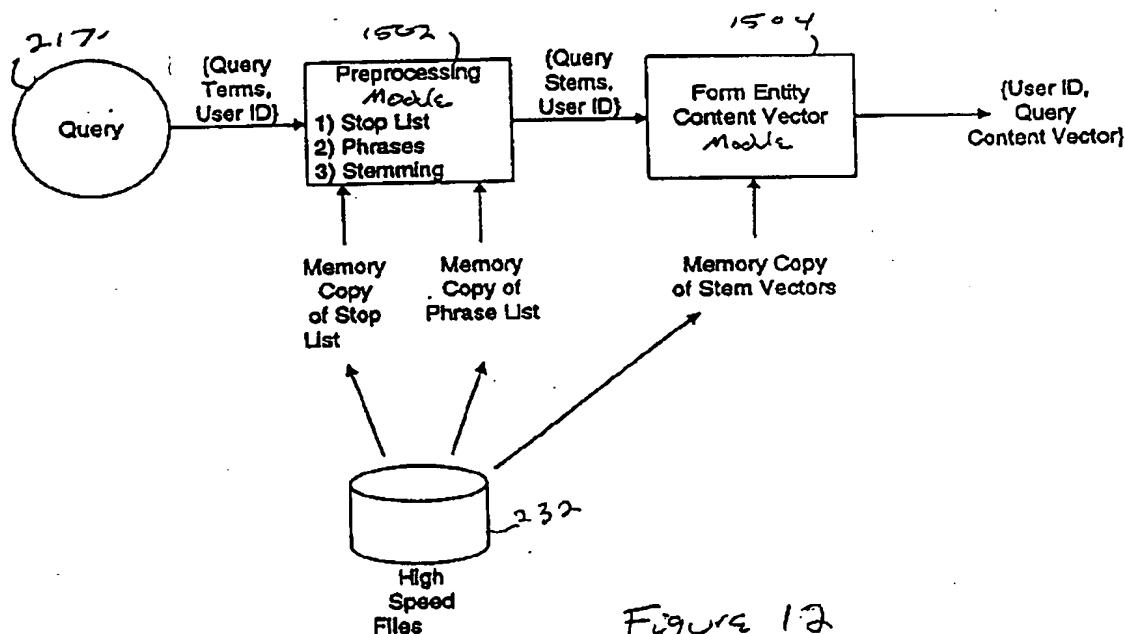


Figure 12

P - 6 1 6 8 F

- 10 -

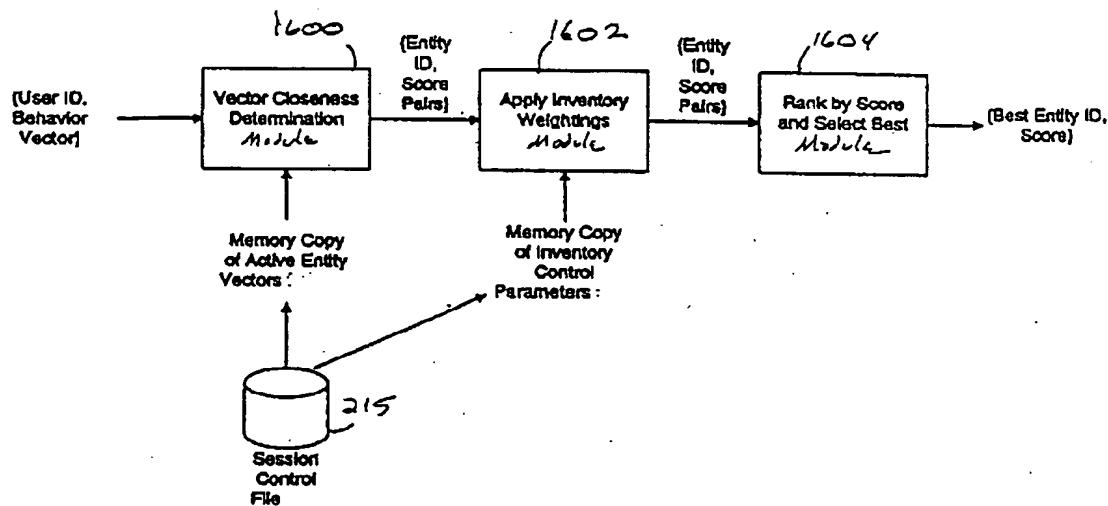


Figure 13

P - 6 1 6 8 F

- 11 -

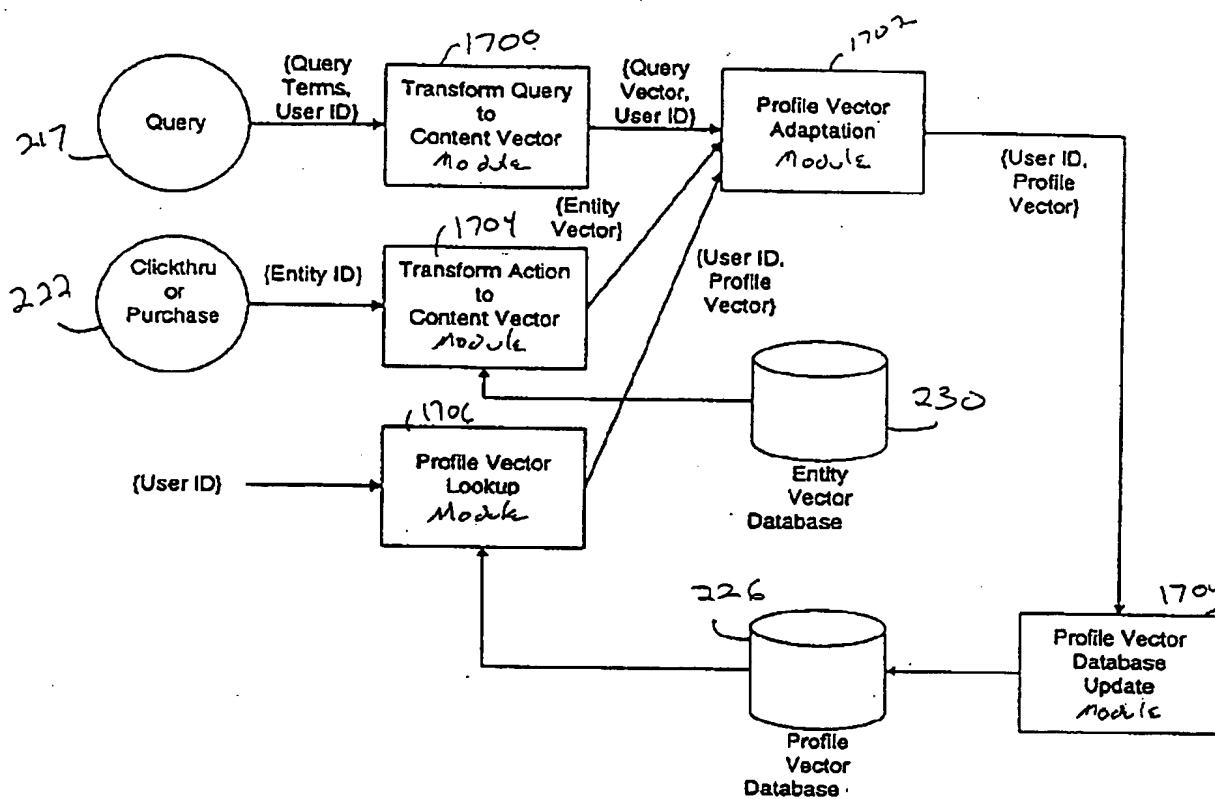


Figure 14

P - 6 1 6 8 F

- 12 -

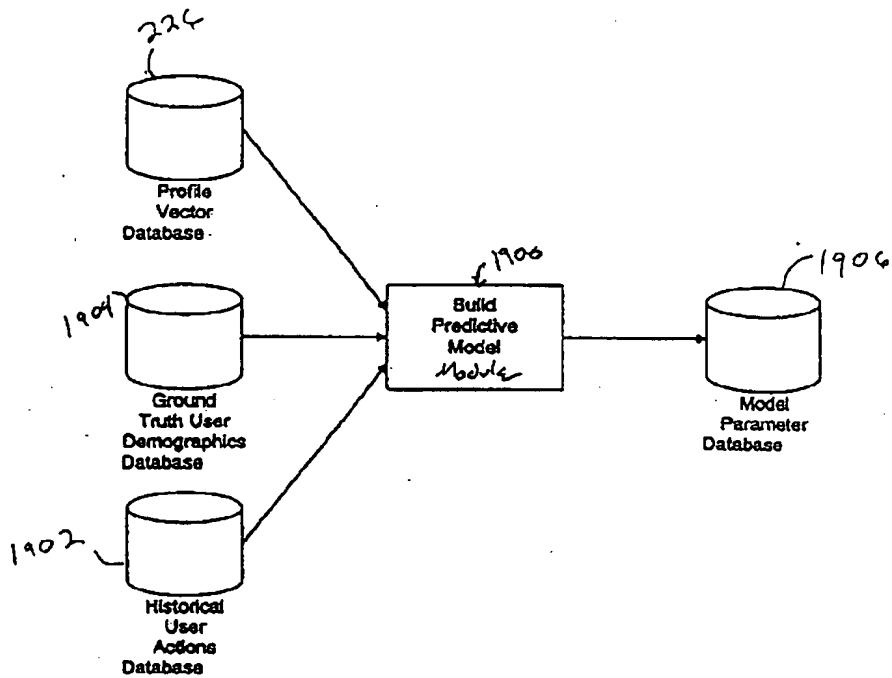


Figure 15

P - 6 1 '6 8 F

- 13 -

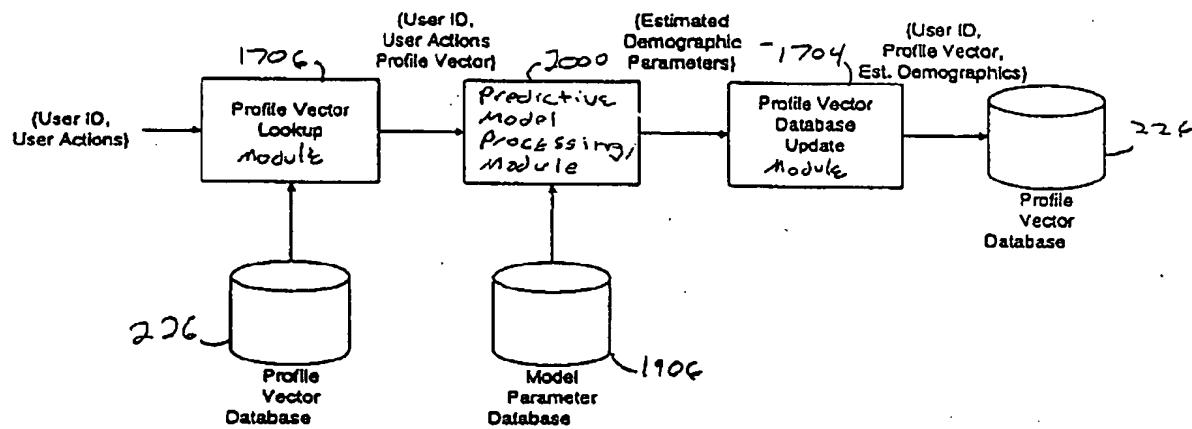


Figure 16

P - 6 1 6 8 F

- 14 -

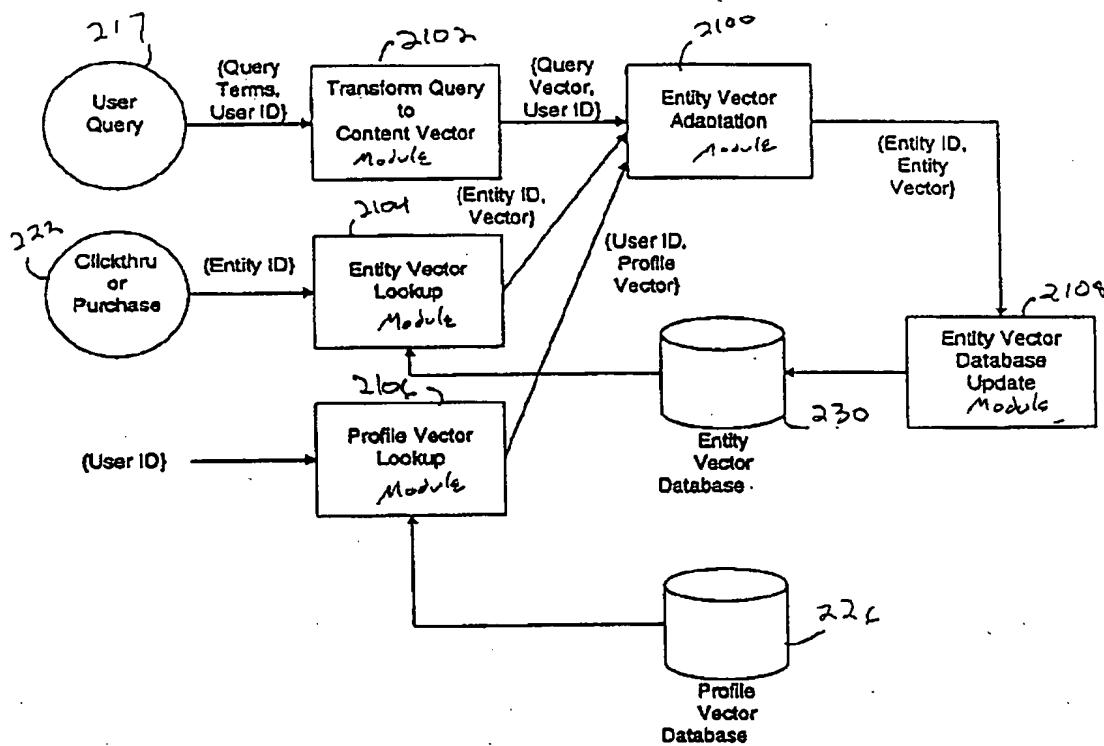


Figure 17

P - 6168 F

- 15 -

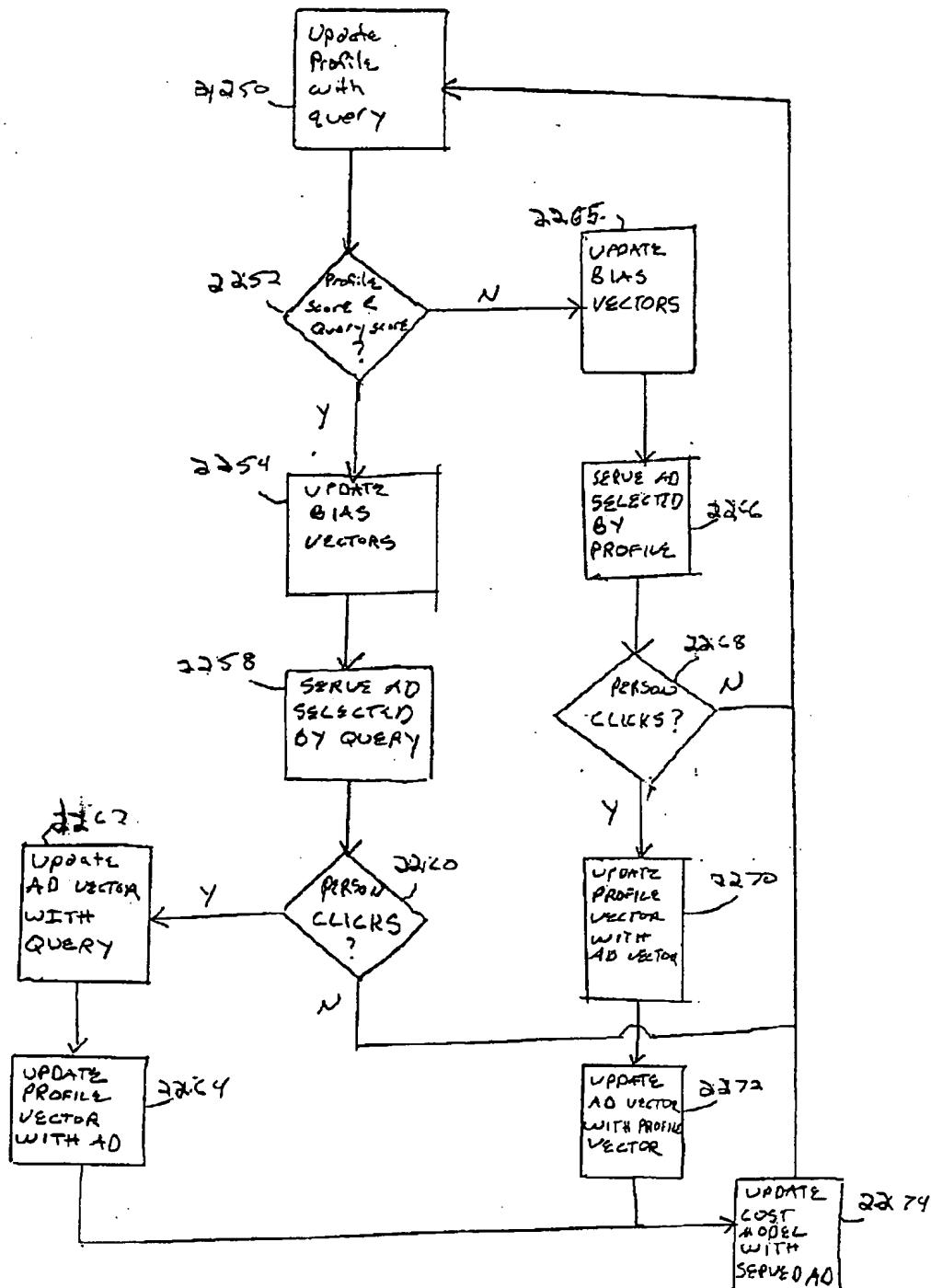


FIGURE 18

P - 6 1 6 8 F

- 16 -

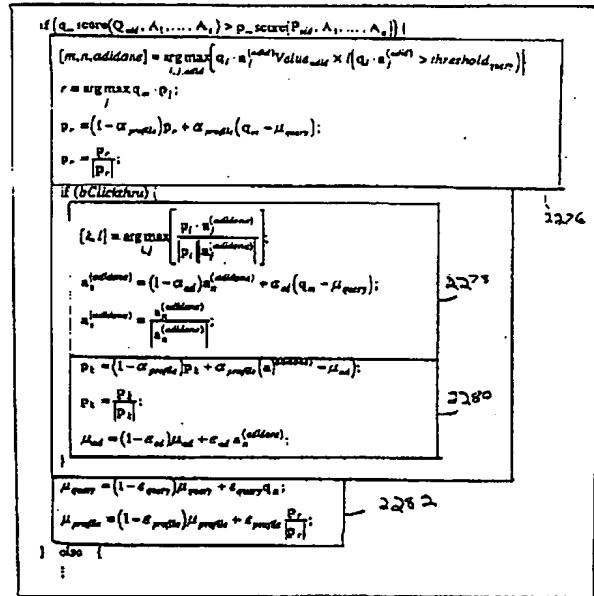


Figure 19 a

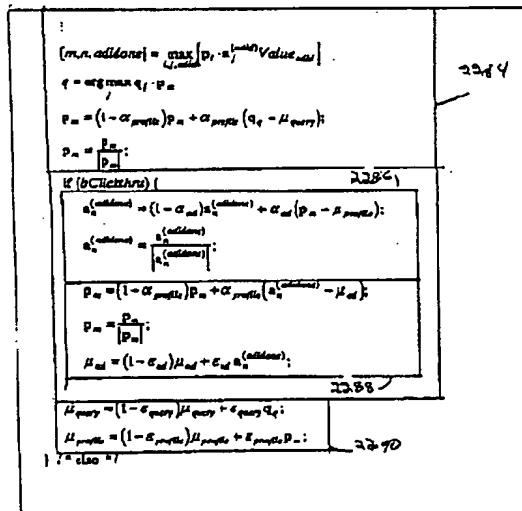


Figure 19 b

P - 6 1 6 8 F

- 17 -

Name	Typical Value	Value Range	Description
$\text{threshold}_{\text{query}}$	0.5	0.25-0.75	Threshold by which query vector will be used instead of profile vector.
α_{profile}	0.1	0.05-0.5	Learning rate for profile update.
α_{ad}	0.01	0.005-0.1	Learning rate for ad update.
$\varepsilon_{\text{query}}$	1.0e-06	1.0e-05 - 1.0e-07	Update rate for query universe estimate of the mean.
$\varepsilon_{\text{profile}}$	1.0e-06	1.0e-05 - 1.0e-07	Update rate for profile universe estimate of the mean.
ε_{ad}	2.5e-05	1.0e-04 - 1.0e-06	Update rate for ad universe estimate of the mean.
Δ_{profile}	0.01	0.001 - 0.5	Forgetting factor for profiles
Δ_{ad}	0.01	0.001 - 0.5	Forgetting factors for ads.
μ_{ad}	0	Application dependent	Mean of ads.
μ_{profile}	0	Application dependent	Mean of profiles..
μ_{query}	0	Application dependent	Mean of queries.

FIGURE 20

1 Abstract

A system and method for selecting and presenting personally targeted entities such as advertising, coupons, products and information content, based on tracking observed behavior on a user-by-user basis and utilizing an adaptive vector space representation for both information and behavior. The system matches users to entities in a manner that improves with increased operation and observation of user behavior. User behavior and entities (ads, coupons, products) and information (text) are all represented as content vectors in a unified vector space. The system is based on an information representation called content vectors that utilizes a constrained self organization learning technique to learn the relationships between symbols (typically words in unstructured text). Users and entities are each represented as content vectors.

2 Representative Drawing Fig. 1